



TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

JANI OLMALA
KETTERÄ PROJEKTIHALLINTA BI-PROJEKTISSA

Diplomityö

Tarkastaja: professori Hannu Jaakola
Tarkastaja ja aihe hyväksytty
TTY:n tuotantotalouden ja rakentamisen tiedekunnassa 24. huhtikuuta 2017

TIIVISTELMÄ

JANI OLMALA: Ketterä projektinhallinta BI-projektissa

Tampereen teknillinen yliopisto Porin yksikkö

Diplomityö, 75 sivua, 0 liitesivua

Kesäkuu 2017

Johtamisen ja tietotekniikan diplomi-insinöörin tutkinto-ohjelma

Pääaine: Ohjelmistotuotanto ja tiedonhallinta

Tarkastaja: professori Hannu Jaakkola

Avainsanat: Business Intelligence, liiketoimintatiedon hallinta, scrum, projektinhallintamenetelmä, ketterä menetelmä

Lähtökohta tutkimukselle oli tutkia, että soveltuuko ketterä prosessimalli BI-projektin kehitysprojektin prosessimalliksi ja lähinnä etsiä mitä tekijöitä täytyy ottaa huomioon perinteisen BI-projektin ja alati nopeasti muuttuvan liiketoimintaympäristön näkökulmasta. Läpinäkyvyys, kommunikoinnin paraneminen, tiimin sitoutuminen, priorisoidut työtehtävät sekä iteratiivisuus ja nopea arvontuotto asiakkaalle voitaisiin lukea etuihin. Myös tiimin ja asiakkaan jaettu vastuu sekä systemaattiset toimintamallit auttavat haluttujen tavoitteiden saavuttamisessa. Menetelmän tulee olla myös joustava, jotta sitä voidaan hyödyntää eri laajuisissa asiakasprojekteissa.

Tiettyjä esivalmisteluja vaativan teknisen ympäristön ja mahdollisesti monitoimittajaympäristön hallitsemiseksi on tarpeen saavuttaa ketterä projektinhallintamalli. Voitaisiin ajatella myös useammassa kehityspotkussa kulkevien iteraatioiden tarpeellisuudesta, jossa jokaisella kehitystiimillä ja tiiminjäsenellä on oma vastuualueensa. Arkkitehtuurin suunnitteluvaiheessa herää myös kysymys, että saadaanko kaikki tehtäväkokonaisuudet ositettua riittävän pieniin ja hallittaviin tehtäviin? Ratkaisun kehittämisen läpinäkyvyys ja muutostenhallinta paranevat, kun asiakas ja tuotteenomistaja ovat sitoutuneet menetelmään. Vähimmilläänkin menetelmän voisi ajatella sopivan BI-järjestelmän ylläpito- ja pienkehitystöihin sekä hallitun kokoisiin kokonaisratkaisun toimituksiin. Jatkuva toiminnan kehittäminen retrospektiivin avulla on myös selkeä hyöty, koska iterointi tapahtuu riittävän nopeissa sykleissä ja kannustaa tiimejä sekä yksilöitä jatkuvan kehittämisen periaatteeseen. Mielenkiintoinen ja laaja aihe, josta on ollut tavoitteena löytää oikeat vaiheet menestyksekkääseen projektinhallintaan BI-projekteissa.

ABSTRACT

JANI OLMALA: Agile Project Management in BI-project

Tampere University of technology

Master of Science Thesis, 75 pages, 0 Appendix pages

June 2017

Master's Degree Programme in Management and Information Technology

Major: Software Engineering and Data Management

Examiner: Professor Hannu Jaakkola

Keywords: Business Intelligence, Scrum, Project Management method, Agile method

The starting point for the study was to examine whether agile process model is suitable for the process model of BI-project development, and more specifically, to research the factors that need to be considered from the perspective of a traditional BI-project and the constantly changing business environment. Transparency, improved communication, higher level of commitment from team members, prioritized tasks, iteration processes and early delivery business value for the customer could be seen as beneficial outcomes. Furthermore, the shared responsibility between the team and the customer, and systematic operation models facilitate the achievement of the desired goals. The method needs to be flexible in order to be able to apply it to customer projects of varying sizes.

An agile project management method is important to establish in order to manage a technical multiple supplier environment that requires specific preparation work. The need for different iterations proceeding in multiple developmental paths, where each development team and team member has their own areas of responsibility, can also be considered. In the planning stages of the architecture, the question of whether all projects can be divided into appropriately small and manageable tasks, is also raised. Transparency of solution development and change management are also improved when the customer and the product owner have committed to the method. At minimum, one could foresee that the method would be suitable for the maintenance and small developmental operations of the BI-system and for the deliveries of manageable-sized complete solutions. Consistent operations development through a retrospective approach is also a clear benefit, as iteration will take place in sufficiently frequent cycles, therefore encouraging the teams and individuals to follow the principle of continuous development. This is an interesting and broad topic to examine in order to find the correct phases for successful project management in BI-projects.

ALKUSANAT

Diplomityön synnyttäminen ja valitun aiheen ympärillä pohtiminen oli minulle miellyttävä prosessi, sillä työelämäkokemusta ohjelmistokehityksen ja –ratkaisujen parissa on kertynyt vajaa 20 vuotta. Se on synnyttänyt minulle käsityksen, kuinka haluaisin itseäni johdettavan ja kuinka asiakas sekä projektitiimi voisivat toimia yhdessä kuin työtoverit ilman suuria raja-aitoja mutta kuitenkin niin, että yhteistyö on päämäärätietoista ja tuloksia tuottavaa. Diplomityön aihe syntyi tieteellisen kirjoittamisen kurssilla löytyneen mielenkiintoisen ja työpaikalleni ajankohtaisen aiheen myötä. Kiitos Harri Kedolle kannustuksesta aiheen parissa. Kurssin ja kannustuksen jälkeen pidin aihetta itselleni sopivana ja siitä lähtien olen mielessäni tutkimusta valmistellut sekä tasaisin aikavälein kirjoittanut tutkimuksen teoriaosuutta vieden tutkimusta pala palalta maaliin. Samalla työnantajallani alkoi ”AgileBI”-termin alla ketterien menetelmien kehitys osaksi laatukäsikirjaa ja projektinhallintaa, jolloin otin asian esille ja päädyimme lopullisesti tähän aiheeseen. Aiheen parissa oli mukava työskennellä ja sain työkavereilta hyvää tukea prosessin kuluessa. Pidin työpaikallani myös koulutuksen tutkimuksen tuloksista.

Haluan kiittää professori Hannu Jaakkolaa, joka on tukenut Diplomityöni etenemistä syyskuusta 2016 aloituskokouksesta lähtien. Hannu toi esille alussa muutaman tärkeän näkökohdan, joita olen pitänyt työssäni ohjenuorana. Haluan kiittää myös työkavereitani pitkistä keskusteluista ja annetuista haastatteluista miettiessäni toimivaa mallia työpaikallamme projektinhallinnantyökaluksi. Lisäksi haluan kiittää rakasta vaimoani Maria sekä kolmea ihanaa tyttöäni Annaa, Hetaa ja Viiviä heidän kärsivällisestä suhtautumisesta hajamieliseen perheenjäseneseen, johon ei aina halutessaan saa yhteyttä, kun asiat etenevät omaa matkaansa mielen pyörteissä.

Oulussa, 28.06.2017

Jani Olmala

SISÄLLYSLUETTELO

1.	JOHDANTO	1
1.1	Tutkimuksen tavoite ja rajaukset.....	1
1.2	Tutkimusmenetelmä.....	3
1.3	Tutkimuksen tulokset ja rakenne.....	4
2.	BUSINESS INTELLIGENCE	7
2.1	BI-projektin elinkaari	8
2.2	BI-arkkitehtuuri.....	10
3.	PROJEKTINHALLINTAMENETELMÄT.....	13
3.1	Projektimallin valinta	13
3.2	Vesiputousmalli.....	17
3.2.1	Vesiputousmallin luonteenpiirteet ja edut.....	17
3.2.2	Vesiputousmallin harhaluuloja ja haasteita.....	19
3.3	Ketterät menetelmät	21
3.3.1	Ketterät periaatteet	21
3.3.2	Ketterien menetelmien vahvuuksia.....	23
3.3.3	Scrum viitekehys.....	27
3.4	Scaled Agile Framework – SAFe.....	30
3.5	Disciplined Agile Delivery - DAD.....	34
3.5.1	DAD-menetelmä ja päämäärät.....	34
3.5.2	DAD vastuut ja roolit.....	39
3.6	Scrum BI-projektin viitekehystenä.....	41
3.6.1	Ketterän BI-projektin edellytyksiä.....	42
3.6.2	Water-Scrum-Fall.....	43
3.7	Mitä etuja ketteristä menetelmistä on BI-hankkeissa.....	45
4.	TUTKIMUS	46
4.1	Kohdeyritys	46
4.2	Tutkimusongelma.....	47
5.	TYÖN TULOKSET JA HAVAINNOT	48
5.1	Projektin aloitus.....	50
5.2	Määrittely	52
5.3	Iteraatiot	62
5.4	Testit ja hyväksyntä.....	65
5.5	Julkaisu.....	66
5.6	Projektin päätös	66
5.7	Artefaktit	67
6.	POHDINTA JA YHTEENVETO	69
	LÄHTEET.....	71

KUVALUETTELO

Kuva 1.	<i>Tutkimuksen vaiheet</i>	4
Kuva 2.	<i>Kimballin elinkaarimalli</i>	10
Kuva 3.	<i>Neljä arkkitehtuurikategoriaa</i>	10
Kuva 4.	<i>Korkeantason DW/BI-järjestelmän arkkitehtuurimalli</i>	11
Kuva 5.	<i>Kolmen dimension projektitavoite</i>	13
Kuva 6.	<i>Projektijohtamisen elinkaaren mallinvalintaprosessi</i>	14
Kuva 7.	<i>Mallien lähestymistavat</i>	15
Kuva 8.	<i>Kehitysvaiheen riippuvuudet</i>	15
Kuva 9.	<i>Vesiputousmalli</i>	17
Kuva 10.	<i>Scrumin kehitysiteraation rakenne ja vaiheiden kesto</i>	28
Kuva 11.	<i>Julkaisun suunnittelukokouksen syötteet ja tulokset</i>	29
Kuva 12.	<i>Hankkeen salkunhallinnassa hyödynnetään ketterää ja vesiputousmenetelmää</i>	31
Kuva 13.	<i>Arvovirta</i>	32
Kuva 14.	<i>Hanketaso</i>	33
Kuva 15.	<i>Tiimitaso</i>	34
Kuva 16.	<i>DAD advanced/lean elinkaari</i>	35
Kuva 17.	<i>DA elinkaaren vaiheet</i>	36
Kuva 18.	<i>Alustusvaiheen tavoitteet</i>	37
Kuva 19.	<i>Rakennusvaiheen tavoitteet</i>	38
Kuva 20.	<i>DA elinkaaren laajennetut vaiheet</i>	38
Kuva 21.	<i>DAD roolit</i>	39
Kuva 22.	<i>Water-scrum-fall yhdistetty elinkaarimalli</i>	43
Kuva 23.	<i>Ketterän kehityksen elinkaari</i>	50
Kuva 24.	<i>Agile-valmius</i>	51
Kuva 25.	<i>Määrittelyvaihe</i>	54
Kuva 26.	<i>Iteraation suunnittelutyöpaja</i>	56
Kuva 27.	<i>Projektitiimin roolit, vastuut ja velvollisuudet</i>	60
Kuva 28.	<i>Projektiorganisaatio</i>	61
Kuva 29.	<i>Iteraatiot</i>	63
Kuva 30.	<i>Iteraation näkökulmia BI-kehityksessä</i>	64
Kuva 31.	<i>Testausprosessi</i>	65
Kuva 32.	<i>Iteraation katselmointityöpaja</i>	67

1. JOHDANTO

Perinteisesti BI-projekteissa on vuosituhannen alussa käytetty vesiputousmallia ja kokemuksen saatossa on herännyt ajatus, että ratkaisut voitaisiin tuottaa nopeammalla aikataululla, paremmalla laadulla sekä edullisemmalla budjetilla. Liiketoimintaympäristön nopeiden muutosten vuoksi myös uusien ominaisuuksien sekä muutostenhallinnan kautta joustavampi projektinhallintamalli olisi tarpeen. Ohjelmistokehityksessä voimakkaasti viimeisen kymmenen vuoden aikana yleistyneet ketterät menetelmät tukisivat edellä mainittua liiketoiminnan tarvetta. Työn tarkoituksena on myös selvittää miltä osin esimerkiksi Scrum-menetelmää ja muita ketteriä menetelmiä voitaisiin suoraan tai soveltaen hyödyntää liiketoimintatiedon hallinnan projekteissa sen sisällöstä ja laajuudesta riippumatta.

Olen sertifioitu ScrumMaster ja käynyt myös muita projektinhallintaan sekä johtamiseen liittyviä kursseja ja koulutuksia. Aikaisemman kokemukseni kautta ohjelmistoprojekteissa Scrumista jäi erinomaiset kokemukset ja tulokset menetelmästä tai metodologiasta. BI-alueen nopea työkalukehitys pakottaa myös alalla toimivat yritykset sopeutumaan nopeampiin läpimenoaikoihin. Kohdeyritykseni tyypilliset toimeksiannot saatavat kestää yhdestä päivästä noin kahteen vuoteen. Näin ollen olemme asettaneet tavoitteeksi pyrkiä tunnistamaan kriittiset sekä tarpeelliset tai pakolliset vaiheet projektinhallintaan sekä projektin koko luokasta riippuen vaihtoehtoisia vaihteita.

Kohdeyrityksellä on myös asiakkaita, joilla ei ole käyteenotettua projektimallia, jolloin voimme sen heille jo tarjousvaiheessa esittää. Projektin kustannukset, aikataulu sekä laadulliset tekijät ovat myös kilpailuvaltteja, jotka nykyisessä kilpaillussa toimintaympäristössä ovat välttämättömiä. Vesiputousmallin tai muun suunnittelulähtöisen mallin puute on myös tunnistettu haaste, että asiakas ei aina tiedä mitä haluaa vaan asiakkaan tarve lähdetään lyhyillä iteraatioilla selvittämään, jolloin muutostenhallinnan täytyy olla joustavaa ja kehitykseen tarvitaan jonkin verran prototypointia tai demokatselmuksia. Tutkimuksen merkitys kohdeyrityksen kilpailuympäristössä on merkittävä, jotta tarvittava kilpailuetu säilytetään. Kohdeyrityksen visiona onkin, että toimitetaan enemmän mitä on luvattu.

1.1 Tutkimuksen tavoite ja rajaukset

Tutkimuksessa tavoitteena on selvittää ja kehittää liiketoimintatiedon hallinnan projektien tueksi ketterä projektinhallintamenetelmä. Tutkimuksen tavoitteena on selvittää, että miltä osin ketterät menetelmät toimivat BI-projekteissa. Menetelmän valinnan kannalta on projektin aloitusvaiheessa myös tarpeen tunnistaa - mitä etua ketterillä menetelmillä saavutetaan suhteessa perinteisiin malleihin. Tutkimuksen tuloksena on tavoit-

teena tuottaa malli, joka voisi tukea myöskin eri kokoluokan projekteja. Projekteissa voidaan esimerkiksi toteuttaa tietovarasto, tiedonsiirrot sekä raportointikerros, kun taas joissakin projekteissa luetaan raportointityökaluun suoraan lähdejärjestelmästä tai tiedostosta. On tarpeen myös selvittää, miten projektien menestymisen kannalta hallitaan suunnittelu, johtaminen, liiketoiminnan vaatimustenhallinta, teknologia, data, ylläpito sekä jatkokehitys.

Tutkimus rajataan käsittelemään projektinhallintaan sekä johtamiseen liittyviä kysymyksiä. Kuinka täytetään asiakastarpeet alkaen vaatimusmäärittelystä ja kattaen myös jatkuvan kehittymisen eli retrospektiivivaiheen. Nämä vaiheet ovat keskeisiä projektien onnistumisen kannalta. Projekteissa tarvitaan myös teknologisesti sekä tiedollisesti osaavat resurssit, sitoutunut omistaja ja sponsori.

Tutkimuksessa keskitytään kohdeyrityksessä pääosin käytössä oleviin prosessimalleihin. Näitä ovat vesiputousmalli ja Scrum-menetelmä sekä asiakkailla käytössä oleva SAFe-menetelmä. Muita prosessimalleja voidaan huomioda työssä kevyemmällä otteella, kuten DAD-menetelmä.

Aikaisempi tutkimus

Ketterästä projektinhallinnasta löytyy lukematon määrä kirjoja sekä artikkeleita. Hiukan suppeammin malleja on käsitelty BI-projektien näkökulmasta. Tavoitteena on tunnistaa tutkimuksen kannalta oleellinen aineisto.

Jurizek (2014) esittelee ketterän projektinhallinnan periaatteita artikkelissaan Agile Project Management Principles. Alussa hän vertaa eroavaisuuksia vesiputousmallin ja ketterien menetelmien periaatteista. Ketterien menetelmien periaatteisiin on artikkelissa lueteltu aktiivinen käyttäjän osallistaminen, vastuullinen tiiminjohtaminen alhaalta ylöspäin, jatkuva vaatimustenhallinta, nopeat ja pienet inkrementaaliset julkaisut sekä iteraatiot ja yhteistyö kaikkien sidosryhmien välillä.

Fitsilis (2008) teki vertailun artikkelissaan Project Management Knowledge (PMBOK) mukaisesti määriteltyjen hankkeiden hallintaprosessien yleisistä vaiheista verraten niitä ketteriin projektinhallintaprosesseihin. PMBOK on Project Management Institute:n kehittämä malli ja se koostuu viidestä prosessivaiheesta (käynnistäminen, suunnittelu, toteutus, valvonta ja päätös). Mallissa on kuvattu myöskin yhdeksän eri osaamisaluetta, joita ovat integraationhallinta, toiminnanhallinta, ajanhallinta, kustannustenhallinta, laadunhallinta, henkilöstöhallinto, kommunikaationhallinta, riskienhallinta ja hankintojenhallinta. Toisaalta ketterä ohjelmistoprojektinhallinta perustuu seuraaviin periaatteisiin: muutoksien hyväksyntä, asiakasarvon tuottaminen, ominaisuuksien toimittaminen inkrementeissä ja yhteistyö sidosryhmien välillä sekä jatkuva oppiminen. Tämän vertailun tarkoituksena oli tunnistaa vertailussa olevien menetelmien puutteita ja eroja. Tuloksena oli, että ketteriä projektinhallintamenetelmiä ei voida pitää täydellisinä perinte-

sen projektinhallinnan näkökulmasta, koska useat prosessit puuttuvat tai niitä ei ole menetelmässä kuvattu.

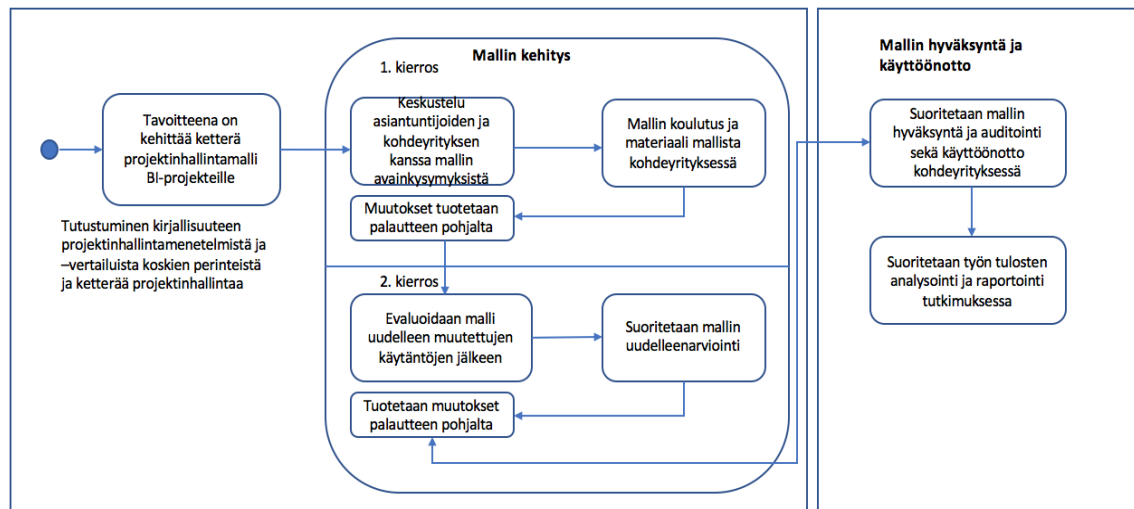
1.2 Tutkimusmenetelmä

Tutkimuksessa käytin Hevnerin et al. (2004) suunnittelutieteellistä lähestymistapaa, koska tavoitteena on tuottaa uusi ketterä projektinhallintamalli, joka sopii luonteeltaan Business Intelligence-projekteihin. Tutkimuksen vaiheet pitävät sisällään pääasiassa ongelman määrittelyn ja artefaktien tavoitteet, suunnittelun ja kehittämisen sekä lopuksi artefaktin katselmoinnin ja hyväksynnän kohdeyrityksessä. Suunnittelutieteen pyrkimyksenä on luoda asioita, jotka palvelevat inhimillisiä tarkoituksia (Simon 1969). Tutkimuksen tuloksia arvioidaan niiden tuottaman hyötyarvon mukaan. Ovatko tuotokset uusia tai ovatko ne parannus entiseen verrattuna? Suunnittelutieteilijät tuottavat ja soveltavat tietoa tehtävistä ja tilanteista luodakseen tehokkaita artefakteja. Suunnittelutieteen lopputulokset ovat konstruktia, malleja, metodeja ja toteutuksia. Kielen käsitteet ja yläkäsitteet voidaan yhdistää ylemmän tason malleiksi, joilla kuvataan tehtäviä, tilanteita ja artefakteja. Suunnittelutieteilijät kehittelevät myös metodeja, tapoja suorittaa tavoitteisia toimintoja. Edellä mainitut voidaan toteuttaa erilaisina tuotteina, jotka on tarkoitettu tiettyä tehtävää varten.

Suunnittelutieteeseen sisältyy kaksi toimintoa: rakentaminen ja arviointi. Rakentaminen tarkoittaa tiettyä tarkoitusta varten luodun artefaktin konstruointiprosessia ja arviointi tarkoittaa arviointiprosessia, kuinka hyvin artefakti palvelee tehtäväänsä.

Suunnittelutieteellisen lähestymistavan kuvaus Järvisen (2008) mukaan.

- Tutkimuksen kohteena ovat ihmisten tuotokset: organisaatiot, tietojärjestelmät. Ihmisten tuotoksia voidaan kehittää, tuottaa ja tutkia.
- Onko tutkija joko ratkaissut nämä ongelmat vai tekeekö parannusta olemassa olevaan ratkaisuun, joka parantaa merkittävästi aiempaa ratkaisua?
- Suunnittelutiede pyrkii tuottamaan ratkaisuja, jotka palvelevat niille asetettuja tarkoituksia, tuotosten hyödyllisyys arvioidaan, ovatko ratkaisumenetelmät uusia? Voidaanko ehdotettuja ratkaisumenetelmiä käyttää muiden kiinnostavien ongelmien ratkaisemiseen?
- Tutkimuksessa toimii vaiheina kehittä ja arvioi.



Kuva 1. Tutkimuksen vaiheet

Kuva 1 kuvaa tutkimuksen yksityiskohtaista menettelytapaa, jota seurasin tässä työssä ehdotetun ketterän projektinhallintamallin kehittämisessä, soveltamisessa ja arvioinnissa. Ensimmäisessä vaiheessa aloitin järjestelmällisellä kirjallisuuskatsauksella, jota seurasi mallin ensimmäinen versio. Toisessa vaiheessa keskustelin kohdeyrityksen asiantuntijoiden kanssa mallin käyttökelpoisuudesta ja mallin jatkokehityksen jälkeen järjestettiin mallin koulutustilaisuus, jossa oli mahdollista asiantuntijoiden antaa palautetta mallista. Koulutus antoi mahdollisuuden keskustella asiantuntijoiden kanssa, joiden arviointi on kriittistä ja sitä voidaan pitää merkittävänä osana artefaktin arviointia. Kolmannessa vaiheessa seurasi artefaktina tuotettu malli, joka hyväksyttiin, auditointi ja käyttöönotettiin asiantuntijoiden toimesta. Tämän jälkeen kirjasin tulokset tämän kirjallisen tutkimuksen tuloksiin ja näin ollen lisäys tietämyskannan malleihin on osaltani tuotettu.

1.3 Tutkimuksen tulokset ja rakenne

Tutkimuksen tuloksena syntyi kohdeyritykselle projektinhallintamalli, joka huomioi kohdeyrityksen tarpeet ja vaatimukset mallin osalta toisistaan erilaisiin hankkeisiin. Menetelmä sisältää kuvauksia parhaista käytännöistä ja vaihtoehtoisia vaiheita projektinhallintaan. Testausprosessin kuvaus syntyi myös tämän tutkimuksen sivutuotteena.

Projektin aloitus

Menetelmässä projektin aloitusvaiheessa analysoidaan asiakkaan ketterien menetelmien valmius. Vaiheessa arvioidaan asiakkaan käytössä olevat järjestelmät ja teknologiat, johtaminen, sopimusmalli, käytettävissä olevat ihmiset ja tilat, prosessit, organisaatio, tavoitteen selkeys ja liiketoiminnan sitoutuminen projektiin.

Määrittely

Määrittelyvaiheessa pidetään asiakkaan kanssa yhteistyössä määrittelytyöpaja. Työpajan tavoitteena on muodostaa ratkaisun visio ja tiekartta. Työpajan tuloksena saadaan aikaiseksi budjetti ja tiiminmääritys. Työpajassa käynnistetään valmisteluiteraatio, jossa työlistä priorisoidaan, suunnitellaan julkaisun aikataulu, ylätasen arkkitehtuuri ja teknologiapäätökset, valmistellaan projektin infrastruktuuri sekä määritellään prosessin yksitiskohdat kuten raportointi, katselmukset ja kommunikointitavat.

Iteraatiot

Vaiheessa toteutetaan halutun lopputuloksen varmistamiseksi tarvittava määrä iteraatioita. Vaatimukset jaetaan tärkeysjärjestyksessä seuraavaksi käynnistymässä olevaan iteraatioon. Tavoitteena on, että iteraation kesto on pisimmillään kalenterikuukausi. Iteraatioiden sisällöt vaihtelevat BI-järjestelmän eri arkkitehtuurin osien kehittämistehtävistä loppukäyttäjälle tarjottaviin visuaalisiin työkaluihin ja osalla näistä tehtävistä saattaa olla riippuvuussuhde, joka huomioidaan iteraatioiden sisällöissä riippuvuudet huomioiden. Iteraatio sisältää seuraavat vaiheet: prototypointi, määrittely ja suunnittelu, toteutus, testaus, hyväksymistestaus ja käyttöönotto.

Testit ja hyväksymistyöpaja

Hyväksymistyöpajassa varmistetaan, että järjestelmän eri osat ja integraatiot toimivat yhdessä tuotantoympäristössä tuotantodatalla. Asiakas hyväksyy ratkaisun ominaisuudet tai palauttaa vaillinaiset ominaisuuden työlistalle.

Julkaisu

Kun hyväksymiskriteerit on kaikilta osin hyväksytty valmiin määritelmän mukaisesti, suoritetaan ratkaisun julkaisu ja käyttöönotto. Käyttöönoton alkuvaiheessa valmistellaan käyttöympäristö kaikilta osin koskien palvelimia, työkaluja ja sisältöä sekä mahdollisia yhteyksiä eri tietojärjestelmiin. Ratkaisu on valmis liiketoiminnan käyttöönotettavaksi tuottamaan arvoa päivittäiseen työhön.

Päätös

Projektin päätösvaiheessa pidetään retrospektiivi, jossa käydään läpi iteraation aikana hyväksi havaitut työtavat ja sovitaan tiimin kesken parannettavat asiat prosessissa. Asiakas voi tilata ratkaisun käyttöön tukea tai koulutuksia. Ratkaisun takuu-aika alkaa ja samalla sovitaan jatkokehitystoimenpiteet ratkaisulle.

Työn rakenne

Tutkimuksen rakenne on jaettu kuuteen lukuun. Luvussa yksi on johdanto, joka esittelee tutkimuksen tavoitteet ja rajauksen sekä käsittelee aiheen aikaisempaa tutkimusta. Joh-

danto esittelee tutkimusmenetelmän, työn tulokset sekä rakenteen. Tutkimus etenee käsittelemällä luvussa kaksi liiketoimintahallintaympäristöä, BI-arkkitehtuuria sekä BI-elinkaarta sekä BI-järjestelmien ominaispiirteitä. Luvussa kolme esitellään ohjelmistokehityksessä käytettyjen prosessimallien teoriaa vesiputousmenetelmän sekä ketterien menetelmien näkökulmasta. Skaalautuvat projektinhallintamenetelmät ketterillä lähtökohdilla toimivat tutkimuksen ohjenuorana. Luvusta 5 eteenpäin esitellään teorian sekä käytännön kokemuksen pohjalta syntyvä malli kohdeyrityksen projektinhallintaan. Luvussa kuusi pohditaan työn tuloksia ja mallin toimintaa peilaten ympäröivään liiketoimintamaailmaan sekä esitellään tutkimuksen aikana syntynyt jatkotutkimuksen aihe.

2. BUSINESS INTELLIGENCE

Seuraavissa kohdissa käydään läpi mitä Business Intelligence on ja mitä BI-projektin läpivieminen pitää yleisesti ottaen sisällään. Liiketoimintatiedon hallinnan merkitys yritysten toiminnan ohjauksessa on merkittävästi kasvanut kuluvin vuosina. Yritysten haasteena on hyödyntää olemassa olevaa suurta tietomäärää, jota löytyy niin yrityksen operatiivisista järjestelmistä kuin mahdollisesti sosiaalisen median tuottaman datan muodossa. Liiketoimintatiedon hallinta tarjoaa edellä mainittuun haasteeseen yhden ratkaisun, jossa tiedot kerätään yhteen paikkaan esimerkiksi tietovarastoon, jotta saavutetaan haluttu tiedon rakenne ja hyödynnettävyys tiedon jalostamiseen.

Hovi et al. (2009) ovat listanneet liiketoimintasyitä ja teknisiä tarpeita miksi tietovarastointi kiinnostaa yrityksiä. Listassa on kuvattu liiketoimintasyitä.

- Tietoresurssien tarkempi analysointi liiketoiminnan kilpailutilanteessa.
- Yritysten elinehtona on hyödyntää asiakastietoja tehokkaammin.
- Yrityksen toiminnasta saadaan aiempaa tuoreempaa informaatiota eri liiketoimintayksiköille.
- Yrityksen tietojen tarjoaminen yhteistyökumppaneille, toimittajille ja tilaajille määriteltyjen käyttörajoitusten rajoissa.
- Liiketoiminnan kiertokulun nopeuttaminen, josta hyvä esimerkki on tilinpäätösraportoinnin tuottaminen tiheämmässä syklissä.
- Yrityssostot, sulautumiset, allianssit muokkaavat yritysrakenteita, jolloin tiedon yhtenäistäminen on tarpeellista.

Simon (2014) luettelee yrityksen BI-järjestelmäkehityksen tiekartassa huomioon otettavia tekijöitä. Tiekarttaan sisällytettävän järjestelmäkehityksen tulisi olla iteratiivinen, inkrementaalinen ja monivaiheinen. Jokaisen tiekartan vaiheen pitäisi sisältää seuraavat asiat.

- Vaiheessa toteutettavat työt sisältävät vähimmäis- ja enimmäisaikataulut.
- Täydellinen luettelo kaikista enakkovaatimuksista, jotka on otettava käyttöön tai olla valmiina ennen kyseisen vaiheen alkamista, mukaan lukien kaikki kriittiset menestystekijät.
- Yksityiskohtainen analyysi kaikista enakkotekijöistä ja liiketoiminta- sekä teknologiavaatimuksista, jotka voivat vaikuttaa kyseisen vaiheen työn onnistumiseen tai epäonnistumiseen.
- Täydellinen luettelo kaikista tuotteista, alustoista, tekniikoista, joita käytetään sekä tarvittava koulutus näille osa-alueille.
- Varasuunnitelma, jos osoittautuu, ettei valitut tiedonhallintatuotteet tai analyysityökalut toimi niin kuin oli ajateltu.

- Täydellinen luettelo kaikista tunnistetuista riskeistä sekä riskienhallintasuunnitelma.
- Tarvittavat henkilöresurssit, joiden roolit ja vastuut ovat selvitetty.
- Perusteellisesti määritellyt kriteerit, joilla mitataan, että vaihe on onnistunut.
- Kokonaisratkaisun- ja projektinhallinnanmallit, toimintatavat ja laajentumissuunnitelmat on analysoitu.

Tiekartan tavoitteena on muodostaa hankkeelle korkeantason näkymä ratkaisun vaatimusten pohjalta pidemmälle aikavälille. Tunnistetaan ratkaisun vaatimukset ja tärkeysjärjestys sekä näiden arvioiden pohjalta laaditaan karkea tuotteen tiekartta. Pidemmissä hankkeissa tiekarttaa tarkistetaan ja päivitetään tarvittaessa vähintään kaksi kertaa vuodessa.

2.1 BI-projektin elinkaari

Mallina BI-projektin elinkaaresta käytän Ralph Kimballin lähestymistapaa. Konseptin ovat alun perin kehittäneet 1980-luvulla Kimball Groupin jäsenet ja useat kollegat Metaphor Computer Systemsistä. Kun he ensimmäisen kerran julkaisivat metodologian The Data Warehouse Lifecycle Toolkit (Wiley, 1998) niin se referoitiin myöskin Business Dimensional Lifecyclestä, koska metodologian nimi sisältää kolme konseptin peruseriaatetta.

Nämä kolme peruseriaatetta olivat:

- Metodologiassa keskitytään lisäämään liiketoiminnan arvoa koko yrityksessä.
- Ajatuksena on, että dimensionaalinen tietomalli tuottaa tiedot liiketoiminnalle raporttien ja kyselyiden kautta.
- Iteratiivisesti kehitetty ratkaisun elinkaari on hallittavampi kuin yrittää kerralla toteuttaa kokonaistoimitusta.

Kimballin (2013) määrittelemät elinkaaren vaiheet on esitetty alla.

Projektinsuunnittelu ja -hallinta

Ensimmäinen vaihe tiekartassa keskittyy saamaan hankkeen tai projektin käynnistettyä sisältäen seuraavat vaiheet - rajauksen, oikeutuksen ja henkilöresurssit. Läpi elinkaaren käytävä suunnittelu hankkeen- tai projektinhallinnan osalta pitää toiminnan raiteillaan.

Liiketoimintavaatimukset

Liiketoiminnan vaatimukset ovat keskeinen osa-alue Kimballin elinkaaresta, koska nämä löydökset ohjaavat eniten kehitysvaiheen päätöksiä. Vaatimukset kerätään määrittämään keskeisiä liiketoimintaan vaikuttavia tekijöitä keskittymällä siihen mitä liiketoiminnan loppukäyttäjä tarvitsee tänään tai tarvitsee tulevaisuudessa mieluummin kuin kysymällä ”mitä haluatte tietovarastolta?”. Suurimmat mahdollisuudet yrityksessä tun-

nistetaan ja priorisoidaan liiketoiminnan saavuttaman arvon ja toteutettavuuden perusteella ja näin yksityiskohtaiset vaatimukset on koottu ensimmäiseen iteraatioon Data Warehouse- tai Business Intelligence-järjestelmän kehityksen kannalta. Kolme samanaikaista elinkaaren rinnakkaista polkua seuraa liiketoiminnan vaatimusten määrittelyn jälkeen.

Teknologia

DW/BI-ympäristöissä saadaan mandaatti useiden teknologioiden integrointiin, tietovarastoihin ja näihin liittyviin metadatoihin. Vaihe alkaa järjestelmän arkkitehtuurin suunnittelulla, jonka myötä vahvistetaan hankittavan ympäristön riittävä kyvykkyys ja määritetään järjestelmä, joka on riittävä kokonaisuuden vaatimuksiin.

Data

Data-vaihe alkaa kohteen dimensionaalisen mallin suunnittelulla, joka täyttää liiketoiminnan vaatimukset ja ottaa huomioon datan asettamat realiteetit. Sana Kimball on synonyymi dimensionaaliselle mallinnukselle, jonka periaatteiden mukaisesti tietomalli mallinnetaan metriikoihin ja dimensioihin. Riippumatta alustasta dimensionaalinen malli pyrkii täyttämään kaksi tavoitetta: helppokäyttöisyys käyttäjän näkökulmasta ja nopea kyselyn suorituskyky. Fyysisen järjestelmän suunnittelu pohjautuu määriteltyyn dimensionalliin, jossa suorituskyvyn mahdollinen parantamistarve otetaan huomioon. Sitten suunnitellaan ETL-prosessi. Kimballin mukaan neljä päätoimintoa ovat tiedon lukeminen lähteestä, tiedon puhdistaminen tai muuntaminen, datan siirtäminen esityskerrokselle ja ETL-ympäristön hallinta.

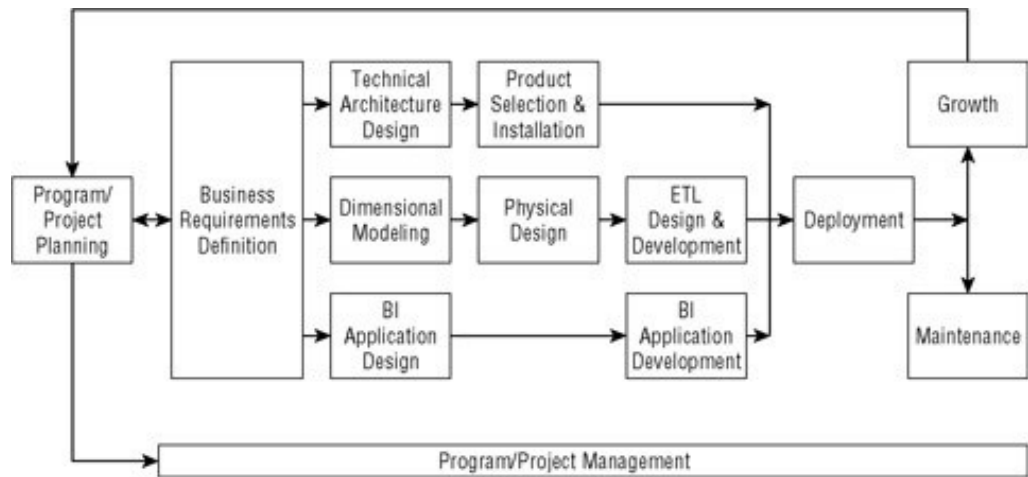
Business Intelligence

Projektin jäsenet keskittyvät työssään ydinosaamisensa mukaan joko teknologioihin ja dataan tai BI-sovelluksiin. BI-sovelluksiin sisältyvät standardoidut raportit, parametrisoidut kyselyt, dashboardit, scorecardit, analyttiset mallit ja tiedon louhintasovellukset.

Julkaisu, ylläpito ja jatkokehitys

Kolme elinkaaren rinnakkaista vaihetta yhdistyvät käyttöönottoon tuoden yhteen teknologian, datan ja BI-sovellukset. Käyttöönoton jälkeen alkaa ylläpitovaihe ja lisäominaisuuksien tarpeiden kautta siirrytään jälleen suunnitteluvaiheeseen.

Edellä mainittujen elinkaaren vaiheiden tehtävät ovat yksityiskohtaisemmin esitetty kuvassa 2.

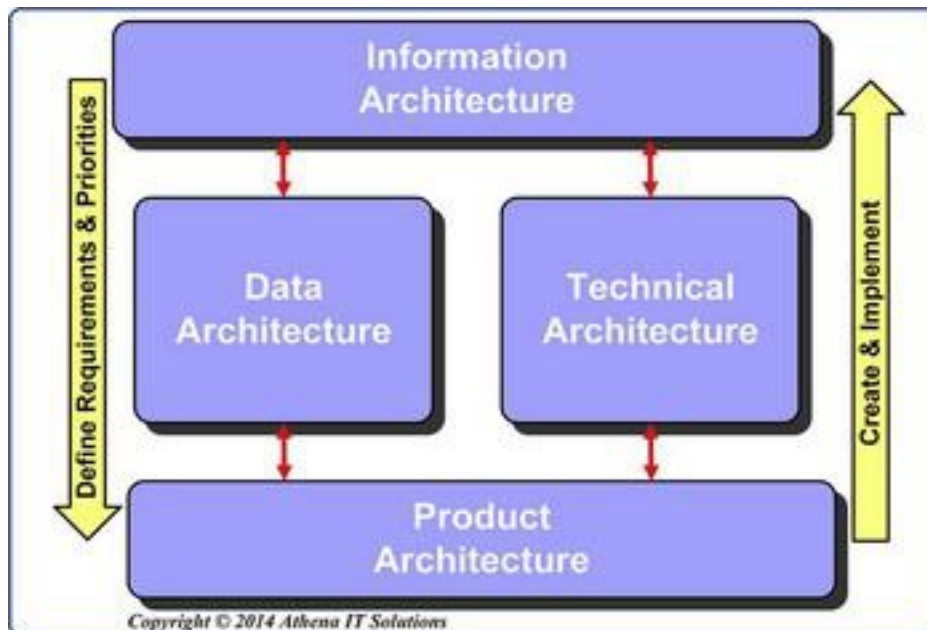


Kuva 2. Kimballin elinkaarimalli

Kaavio kuvaa tehtävien sekvenssiä, riippuvuutta ja rinnakkaisuutta. Se toimii tiekarttana, joka auttaa tiimejä tekemään oikeita asioita oikeaan aikaan. Kaavio ei ole absoluuttinen aikajana, vaikka laatikot ovat samankokoisia. Tehtävien sisällössä on suuria eroja kullekin tehtäväkokonaisuudelle vaadittavasta ajasta ja työstä.

2.2 BI-arkkitehtuuri

BI-arkkitehtuuri voidaan jakaa neljään kategoriaan, joita ovat informaatio-, tekninen, data- ja tuotearkkitehtuuri. Kuvassa 3 esitetty BI-arkkitehtuurikategoriat (Sherman 2015).

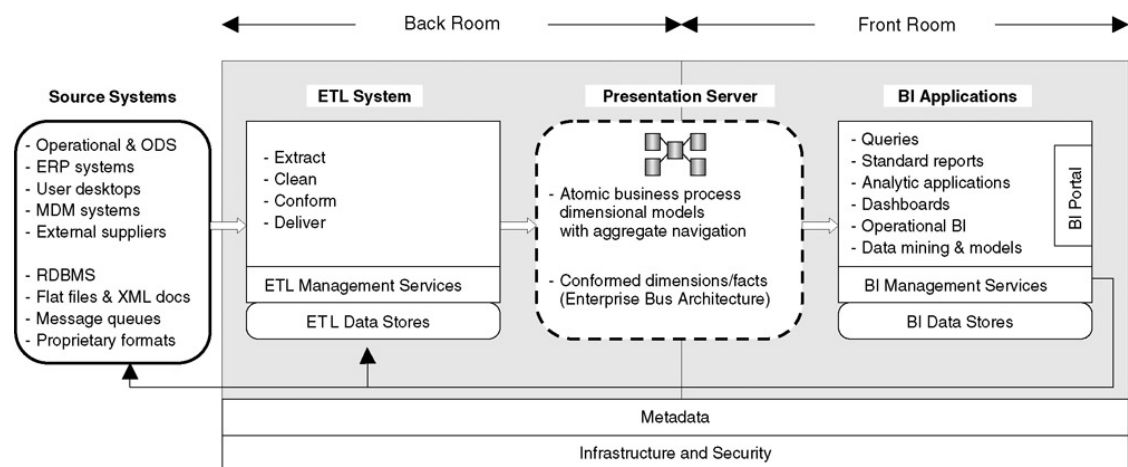


Kuva 3. Neljä arkkitehtuurikategoriaa

Kuvassa 3 kuvatuista kategorioista tässä työssä esitellään tekninen arkkitehtuuri. BI:n teknisessä arkkitehtuurissa on tyypillisesti neljä kerrosta (Kimball 2010, 2013).

- Tiedon lähteenä toimivat operatiiviset järjestelmät. Lähdejärjestelmien tärkeimmät prioriteetit ovat suorituskyky ja käytettävyys.
- ETL-prosessi, jossa data haetaan lähdejärjestelmästä, muokataan ja ladataan tietovarastoon. Tämä tiedon integrointi hallitaan kyvykkäillä integrointiohjelmistoilla.
- Tieto tallennetaan tietovarastoon, josta tiedon raportointi ja analysointi suoritetaan. Tiedot on organisoitu, tallennettu ja ovat käytettävissä loppukäyttäjille eri BI-sovellusten kautta.
- Viimeinen suuri komponentti Kimballin DW/BI-arkkitehtuurissa on BI-sovellukset. Raportointi, analysointi ja tiedon hyödyntäminen tapahtuvat tässä kerroksessa. BI-sovellus viittaa selkeästi liiketoiminta-asiakkaille tarjottaviin kyvykkyyksiin, joilla tietoa hyödynnetään analyyttisen päätöksenteon tukena.

BI-teknisen arkkitehtuurin tavoitteena on tarjota BI-portfolio liiketoiminnan erilaisiin tarpeisiin mahdollistaen liiketoimintatiedon analyyttisen tarkastelun. Järjestelmän tulee auttaa keräämään kaikki oleellinen liiketoimintatieto ja esittää kokonaiskuva yrityksen tilasta. Se auttaa myös selvittämään yrityksen kehityskohteet sekä kertomaan saavutetut tulokset peilaten suunniteltuihin tavoitteisiin. Hyvällä liiketoimintatiedon hallinnalla voidaan saavuttaa selkeä kilpailuetu.



Kuva 4. Korkeantason DW/BI-järjestelmän arkkitehtuurimalli

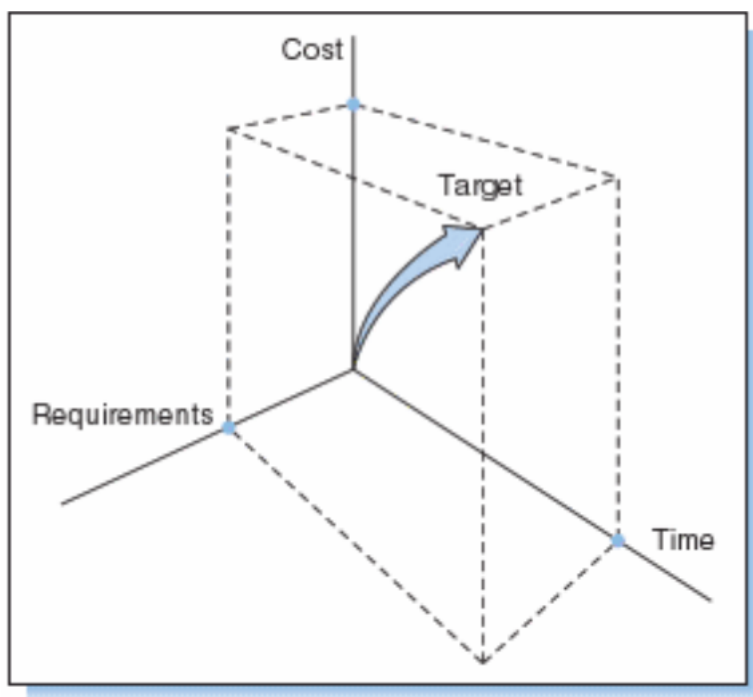
Kuten yllä olevassa kuvassa 4 (Kimball 2008) osoitetaan, niin BI-arkkitehtuuri koostuu useista teknologiakomponenteista. Tiedon erilaisuus, määrä, eheys ja kasvunopeus asettavat omat haasteensa. BI-tuotteen täytyy palvella myös mahdollisesti monia sidosryhmiä, kuten asiakkaat, toimittajat ja yrityksen sisäiset käyttäjät. Onnistuneen BI-järjestelmän kehittämisen kannalta on oleellista myös tietomallin ja teknologioiden sekä työkalujen valinta.

Tieto siirtyy lähdejärjestelmistä ETL-prosessien kautta tietovarastoon, josta tietoa voidaan hyödyntää, esimerkiksi muodostamalla erilaisia raportteja. Hovin (2009) mukaan BI-arkkitehtuuri koostuu seuraavista teknologiakomponenteista:

- BI-ratkaisujen tietolähteinä toimivat yleisimmin yritysten prosessien tukena toimivat operatiiviset, toiminnanohjauksen, asiakkuudenhallinnan ja taloushallinnan järjestelmät.
- Tiedon integrointi toteutetaan ETL-prosessissa (Extract, Transform, Load). ETL-prosessi tietoja haetaan ja muokataan yhdenmukaiseen muotoon sekä ladataan tietovarastoihin. Vaiheessa käytetään tehokkaita integroinnin työkaluja hyödyksi, joilla myös vaihetta pyritään mahdollisimman pitkälle automatisoimaan.
- Tietovarastoon data mallinnetaan sopivaan muotoon raportointia varten. Tietomalleja ovat esim. dimensionaalinen ja Data Vault. Tietovarastoihin ladataan lähdejärjestelmien tieto, laskennat ja summaukset yhtenäiseen muotoon, jotta tietoa voidaan edelleen mahdollisimman helposti hyödyntää raportoinnissa ja analysoinnissa. Tietovarastoinnin etuja ovat tiedon laadun valvonta, pienempi riippuvuus operatiivisista järjestelmistä, historiatiedon tallentaminen ja tiedon nopea hyödynnettävyys.
- Raportointi ja hyödyntäminen ovat BI:n viimeinen vaihe, jossa loppukäyttäjät hyödyntävät liiketoiminnasta kerättyä dataa päätöksenteon tukena. Vaiheessa voidaan datan hyödyntämiseen käyttää erilaisia raportointi- ja analysointityökaluja, jotka mahdollistavat tiedon rajaukset, parametrisoinnit, summaukset, porautumiset, visualisoinnit jne. (Hovi et al. 2009)

3. PROJEKTIHALLINTAMENETELMÄT

Nicholas et al. (2016) esittelevät kolmen dimension mallin, joka esittää eri ulottuvuudet jokaisen projektin tavoitteeksi. Jokaisen projektin tavoitteena on saavuttaa kolme dimensiota, joita ovat kustannukset, aika ja suorituskky. Kustannus on määritelty tai budjetoitu projektille. Projektin käytettävissä oleva aika aikataulutetaan eli missä ajassa työ tulee olla tehtynä. Suorituskky on projektin lopputulos, artefaktit ja se sisältää mitä asiakkaat pitävät tärkeänä tai tarpeellisena.

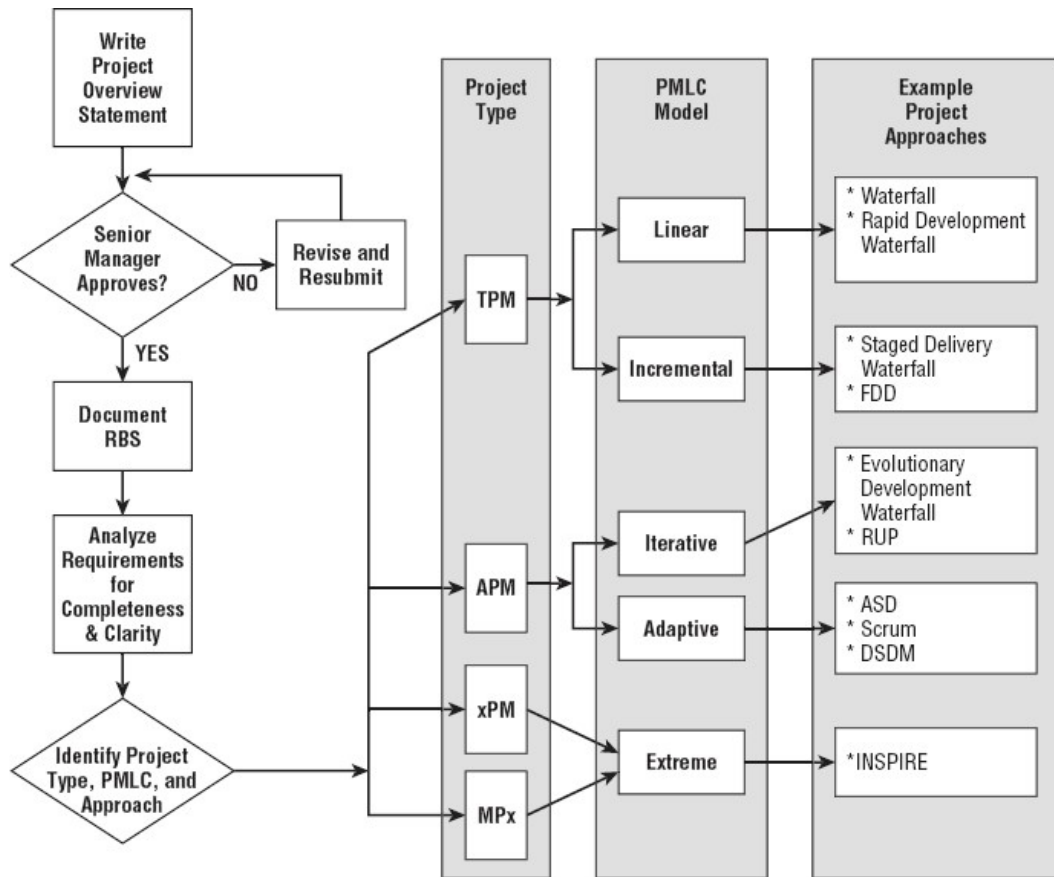


Kuva 5. Kolmen dimension projektitavoite

Tavoite esittää maalia, jossa toimitetaan tiettyinä päivinä ja tietyllä kustannuksella asiakkaan haluamat vaatimukset. Tavoite projektinhallinnalle on saavuttaa projektille asetettu tavoite.

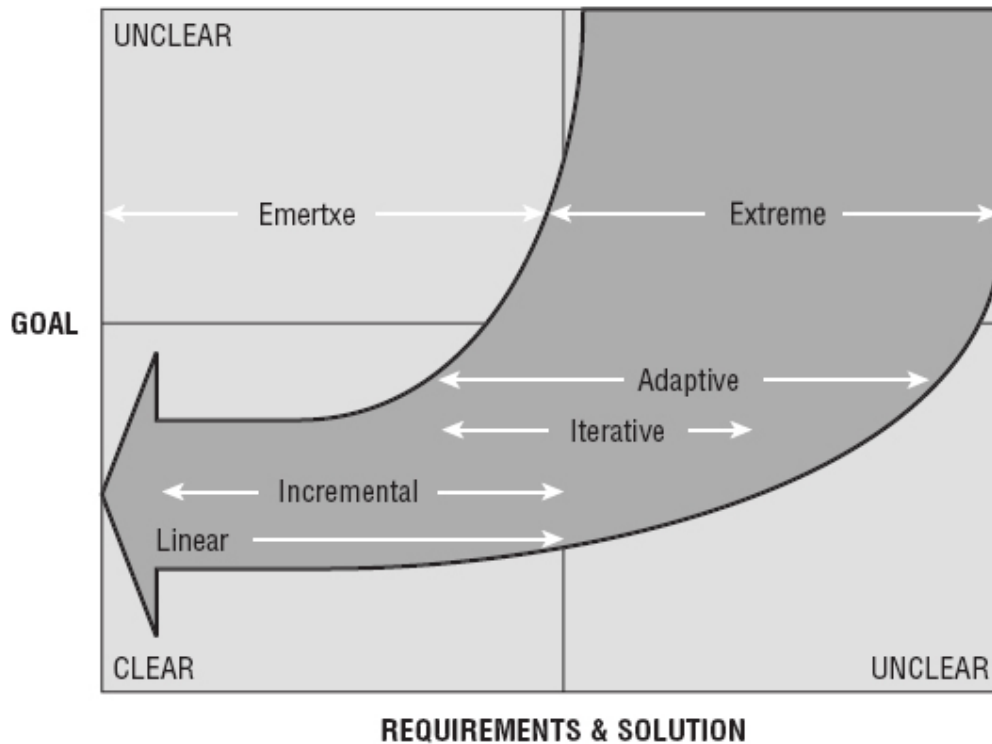
3.1 Projektimallin valinta

Projektin elinkaaren hallintaan valittava malli on subjektiivinen päätös ja se perustuu useisiin muuttujiin. Wysocki (2014) esittelee mallin valintaan prosessin, joka on esitetty kuvassa 6.



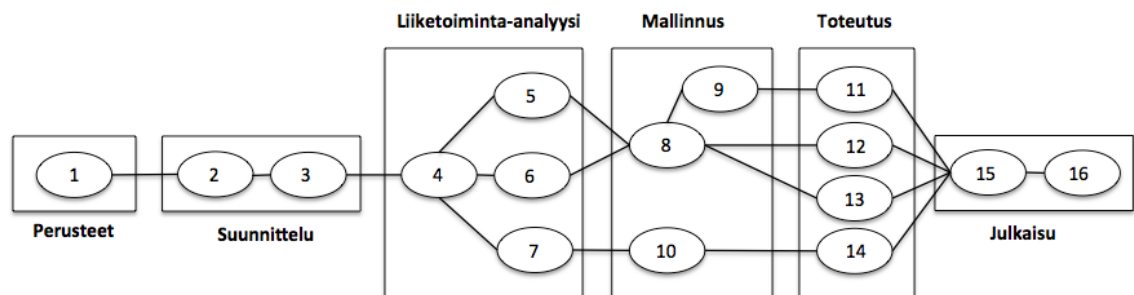
Kuva 6. Projektijohtamisen elinkaaren mallinvalintaprosessi.

Mallin valintaan vaikuttavia muuttujia Wysocki (2014) mainitsee projektin kokonaiskustannukset, kesto, markkinoiden stabiilitetti, teknologia, liiketoiminnan ilmasto, vaikuttavien osastojen määrä, organisaatioympäristö, kehitystiimin taidot ja kyvykkyys. Mallissa esitetietovaatimusten jälkeen on vaihtoehtoina projektityypiksi perinteinen, ketterä, Extreme ja Emertxe lähestymistapa. Seuraavaksi valitaan projektijohtamisen elinkaarimalli, joita ovat lineaarinen, inkrementaalinen, iteratiivinen, mukautuva ja Extreme. Kuvassa 7 on kooste mallien toimivuudesta tavoitteiden sekä vaatimusten ja ratkaisun selkeyden suhteen. Kun nämä vaiheet on käyty läpi, niin viimeisessä osiossa annetaan esimerkkejä projektimenetelmistä.



Kuva 7. Mallien lähestymistavat

BI-projekteissa, kun haetaan ratkaisua liiketoimintatiedon hyödyntämiseen, voidaan käyttää (Moss et al. 2003) esittämää jaottelua, joita ovat oikeutus sekä perusteet hankkeelle, projektin suunnittelu, liiketoiminta-analyysi, mallinnus, toteutus ja ratkaisun julkaisu. Vaiheet on esitetty kuvassa 8. BI-projekti on organisoitu samaan kuuteen vaiheeseen, joita yleisesti käytetään teknisissä projekteissa ja nämä päävaiheet sisältävät projektiin liittyvät yksityiskohtaisemmat kehitysvaiheet.



Kuva 8. Kehitysvaiheen riippuvuudet

Business Intelligence-tiekartta (Moss et al. 2003) kuvailee 16 kehitysvaihetta, jotka sisältyvät näihin projektin yleisiin vaiheisiin. Kehitysvaiheet on kuvattu taulukossa 1. Taulukossa on myös kuvattu kehitysvaiheen vaikutus sen mukaisesti, että vaikuttaako se läpi koko organisaation vai kehitysprojehtin.

Taulukko 1. Projektikohtainen vs. koko organisaatioon vaikuttava

Kehitysvaihe	Vaikutus projekti tai koko organisaatio
1. Liiketoimintavaatimusten arviointi	Organisaatio
2. Yrityksen infrastruktuurin evaluointi	Organisaatio
3. Projektisuunnittelu	Projekti
4. Projektivaatimusten määrittäminen	Projekti
5. Data-analyysi	Organisaatio
6. Sovelluksen prototypointi	Projekti
7. Metatietovaraston analyysi	Projekti
8. Tietokantasuunnittelu	Organisaatio
9. ETL-suunnittelu	Organisaatio
10. Metatietovaraston suunnittelu	Organisaatio
11. ETL-kehitys	Organisaatio
12. Sovelluskehitys	Projekti
13. Tiedonlouhinta	Organisaatio
14. Metatietovaraston kehitys	Organisaatio
15. Julkaisun toteuttaminen	Projekti
16. Julkaisun evaluointi	Organisaatio

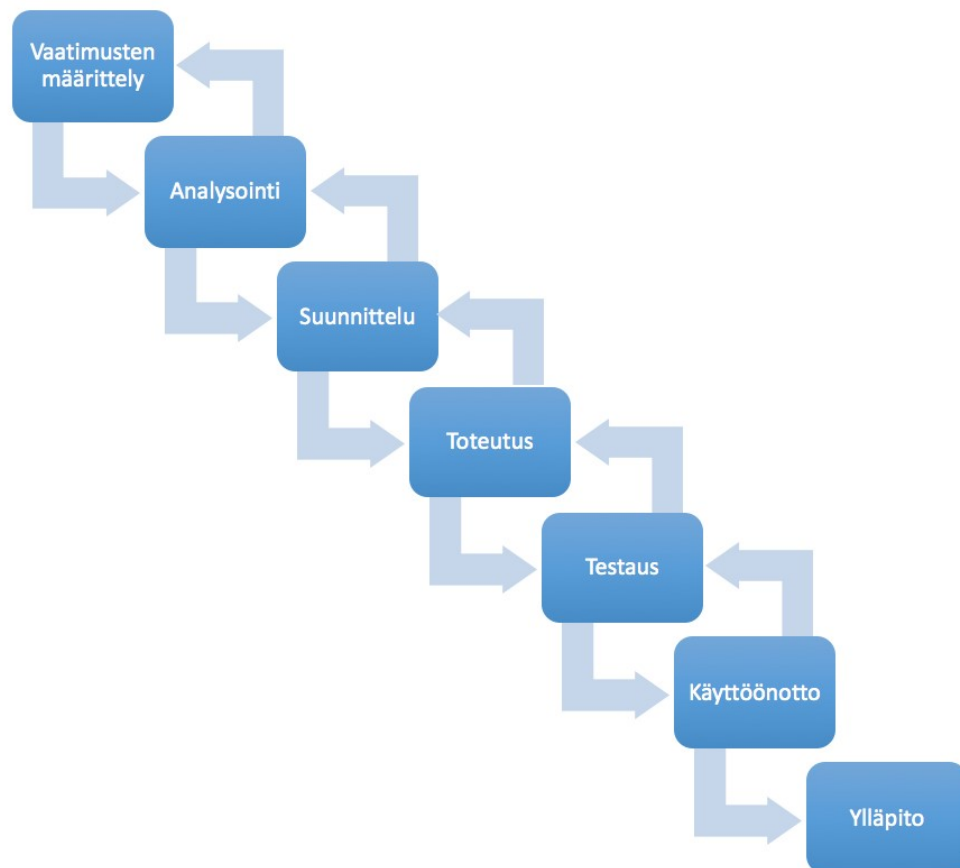
Ensimmäinen projektivaihe on hankkeen olemassaolon oikeutus ja perusteet projektille. Vaiheessa yksi tehdään liiketoimintavaatimusten arviointi. Projektin käynnistämisen mahdollistamiseksi vaiheessa tulisi löytää selkeästi määriteltävät hyödyt yritysongelman ratkaisemisesta tai liiketoimintamahdollisuuksien hyödyntämisestä. Päätös ratkaisun kehittämiseksi tehdään myös kustannushyötyanalyysin tulosten seurauksena. Hankkeen käynnistämiseksi on perusteet, mikäli analyysissä löytyy vähintään yksi tekijä viidestä. Näitä tekijöitä ovat tulojen kasvu, voiton kasvu, asiakastyytyväisyyden kehittäminen, säästöjen lisääminen tai markkinaosuuden kasvattaminen. Vaikka osa kehitysvaiheista on selkeästi projektikohtaisia, useimmat kehitystyöt on suoritettava koko organisaation näkökulmasta. Näissä projektin kehitysvaiheen toiminnoissa pitää ottaa koko organisaation tavoitetila huomioon ja näiden toimintojen tulisi sisältää liiketoiminnan edustajia eri liiketoiminta-alueilta. Liiketoiminta-alueiden asiantuntijoiden pääasiallinen tehtävä on validoida ja ratifioida strategioita, toimintatapoja, liiketoimintasääntöjä ja standardeja, joita joko käytetään tai kehitetään BI-projektin aikana.

Liiketoimintavaatimusten arvioinnin yhteydessä tehdään myös riskianalyysi. Moss et al. (2003) jakaa riskit kuuteen tekijään. Riskit on kategorioitu seuraavasti. Projektissa käytettävät teknologiat, toteutukseen tarvittavien kyvykkyyksien ja prosessin kompleksisuus, komponenttien ja datan integraatiot, organisaation moraalinen tuki, kehitystiimin taidot, asenne ja sitoutuminen sekä taloudellisen sijoituksen tuotto.

3.2 Vesiputousmalli

3.2.1 Vesiputousmallin luonteenpiirteet ja edut

Vesiputousmalli on paljon ohjelmistokehityksessä käytetty lineaarisesti vaiheittain etenevä projektinhallintamalli. Vesiputousmallin vaiheet on esitetty kuvassa 9. 1970 tohtori Winston Royce kirjoitti artikkelin (Royce 1970) Waterfall, jossa hän esitteli mallin suurten ohjelmistojen kehittämiseen.



Kuva 9. Vesiputousmalli

Prosessin vaiheet etenevät järjestyksessä vasemmalta oikealle ja seuraava vaihe aloitetaan vasta kun edellinen on saatu päätökseen eli yksi vaihe on käynnissä tietyllä ajanhetkellä. Leffingwellin ja Widrigin (2003) mukaan laajaa projektia hallinnoidaan jakamalla se loogisiksi osavaiheiksi. Alun perin Roycen malli ymmärrettiin yhdensuunta-

sesti etenevänä mallina. Kuitenkin ajatuksena oli, että tarvittaessa palataan edelliseen vaiheeseen, jos edellisen vaiheen tuotoksessa on puutteita. Vesiputousmallin ensimmäisenä vaiheena tulevat järjestelmävaatimukset ja ohjelmistovaatimukset eli yleisnimitään vaatimustenmäärittely. Vaatimusmäärittelyä seuraa suunnittelu ja yleisesti nähdään, että hyvä suunnittelu elinkaaren alkuvaiheessa johtaa merkittäviin säästöihin projektin loppuvaiheessa. Vesiputousmallissa on kyse suunnittelulähtöisestä mallista. Mallissa lähdetään siitä, että jokainen vaihe on toteutettu onnistuneesti loppuun ja on valmis ennen seuraavaa vaihetta.

Tyypillisen vesiputousmallin vaiheet ovat seuraavat (Sommerville 2016).

- Vaatimusmäärittelyssä järjestelmän vaatimukset tunnistetaan ja kirjataan tiedot käyttäjätarpeiden pohjalta. Vaiheen lopussa syntyy järjestelmäkuvausdokumentti, joka määrittää tulevien vaiheiden sisältöä.
- Analysointi- ja suunnitteluvaiheessa vaatimusmäärittelyssä kiinnitetyt vaatimukset kohdistetaan synnyttävän ratkaisun eri toiminnoille. Tästä syntyy ratkaisun ylätasoin arkkitehtuuri. Ylätasoin arkkitehtuuri määrittää eri toimintojen väliset suhteet.
- Ratkaisun toimintojen toteutus on konkreettista koodausta valituilla teknologioilla. Toteutuksen tuloksena valmistuva ratkaisu testataan aluksi yksikkötesteillä, jolloin varmistetaan, että jokainen ratkaisun osa on määrittelyn kaltainen.
- Integraatiovaiheessa eri ratkaisun osat integroidaan toimivaksi kokonaisuudeksi, toimitettavaksi ratkaisuksi. Kokonaisratkaisu testataan järjestelmätesteillä, jotta toteutettu julkaisu vastaa käyttäjiltä tulleita vaatimuksia.
- Onnistuneiden järjestelmätestien jälkeen ratkaisu toimitetaan ja julkaistaan asiakkaalle tuotantokäyttöön ja ylläpitovaihe sekä tukivaihe alkavat.

Jokaisen vaiheen tuloksena syntyy yksi tai useampi dokumentti, jotka hyväksytetään johtoryhmän kokouksissa. Johtoryhmän hyväksynnän jälkeen voidaan aloittaa seuraava vaihe. Ratkaisun tuotantoon siirron jälkeen ylläpitovaiheessa yleensä löytyy toimintoja, jotka eivät vastaa asiakkaan alkuperäistä vaatimusta. Näiltä osin ratkaisun kehittämistä jatketaan sovitun prosessin mukaisesti. Isoimmat virheet saatetaan siirtää erilliseen uuteen projektiin. (Sommerville 2016)

Roycen alkuperäiseen malliin sisältyi pilottijaksoajatus, jossa tuotetta voitaisiin vielä kehittää, joten iteratiivisuus oli mallissa esillä. (Royce 1970) Royce myös suositteli, että prosessi tehdään kahteen kertaan, joissa erotellaan kriittisiltä osin prototyyppi ja julkaistava lopputulos.

Vesiputousmallikonseptin lähtökohtainen näkökulma on, että analysointivaiheeseen käytetään niin paljon aikaa kuin tarvitaan varmistamaan, että kaikki tieto on suunnittelu ja siitä seuraavissa vaiheissa tiedossa. Tarkoituksena on turvata vaadittujen toimitusten oikeellisuus tulevissa vaiheissa. Tällä tavoitellaan täydellistä luottamusta ja selkeää tiekarttaa, jotta haluttu lopputulos saadaan projektin aikana tuotettua. (Davis et al. 2014)

Davis myös kirjoittaa, että vesiputousmalli konsepti on otettu ensisijaisesti käyttöön sen helpon ymmärrettävyyden vuoksi. Se on helppo ymmärtää sekä seurata loogisen ja vaiheittain etenevän lähestymistavan seurauksena.

Vesiputousmalli on parhaimmillaan yksiselitteisissä ja muuttumattomissa projekteissa. Jotkut organisaatiot haluavat pysyä perinteisessä tai vesiputousmallissa projekteissaan. Jos kaikki toimii hyvin niin miksi muuttaa toimivaa käytäntöä? Tärkeää on tunnistaa, että valittu käytäntö pitää organisaation kilpailukykyisenä nyt ja tulevaisuudessa. Perinteinen lähestymistapa sopii hankkeisiin, joista on tunnistettavissa seuraavia piirteitä. (Davis 2013)

- Kun toimit vakiintuneessa liiketoimintaympäristössä.
- Kun vaatimukset voidaan määritellä tarkalle tasolle.
- Organisaatio on laajalle levinnyt ympäri kaupunkia, maata tai maailmaa.
- Organisaation yrityskulttuuri on jäykkä ja muuttumaton. Johtoryhmä ja projektin jäsenet ovat muutosvastarintaisia, jolloin on vaikea muuttaa asioita.
- Uudet tai kokemattomat tiiminjäsenet, joilla on rajattu tai yksipuolinen osaamisalue.
- Hankkeet, joissa on pieni riski, eivät välttämättä tarvitse joustavuutta.
- Vesiputousmalli toimii, joten organisaatiossa ei ole tarvetta muuttaa mallia.

Vesiputousmalli on helppo käyttää ja johtaa. Prosessi seuraa samaa vaiheittain etenevää mallia projektista toiseen, joten se on helppo omaksua ja ymmärtää. Kehitysryhmä ei tarvitse erityistä koulutusta mallin kannalta. Vesiputousmalli on myös jäykkä malli, jossa jokaisella vaiheella on omat tarkat toimitukset ja katselmoinnit, joten sitä on helppo johtaa ja hallita.

Kuri on pakollinen, koska jokaisella vaiheella on alku- ja päätepiste, jolloin edistymisen jakaminen ja seuranta sidosryhmien kannalta on helppoa. Kehitystiimi keskittyy jokaisessa mallin vaiheessa siihen tarkoitettuun osa-alueeseen. Näin ollen aikatauluriski vähenee.

Dokumentointi on kattavaa. Malli vaatii dokumentaation joka vaiheesta, minkä tuloksena seuraavassa vaiheessa on parempi ymmärrys mitä tulisi tehdä edellisen vaiheen jälkeen. Dokumentointi jättää myös työkalun tuleville projekteille, kuinka asiat tehtiin, ja sidosryhmät voivat tarkastella mitä asioista on sovittu tai mikä logiikka on tietyn toiminnallisuuden sekä vaatimuksen taustalla.

3.2.2 Vesiputousmallin harhaluuloja ja haasteita

Davis esittää yhteenvedon vesiputousmallin harhaluuloista.

- On olemassa sellainen projekti kuin 100% vesiputousmallin projekti.
- Vesiputousmalli on sovitettu malli kaikkiin hankkeisiin.

- Vesiputousmalli vähentää riskiä ja toimitukseen liittyviä epävarmuustekijöitä kaikessa kehitystyössä.

Käytännössä useimmilla vesiputousmallilla tehdyillä hankkeilla on jonkin verran iteraatioita useista eri syistä ja usein luotetaan jonkinlaiseen esitutkimukseen tai prototyyppiin. Jotta vesiputousmalli lähestyy reaaliaikaisen tarpeita, on kussakin vaiheessa sisällytettävä toivotun ratkaisun saamiseksi asiakkaan palaute, jotta kaikki käyttäjätarpeet saadaan dokumentoitua täsmällisesti liiketoiminnan tarpeiden mukaan. On epätodennäköistä, että kaikki käyttäjätarpeet saadaan ensimmäisellä kerralla dokumentoitua, kehitettyä ja integroitua tilaajan haluamalla tavalla.

Vesiputousmallin etuina usein koetut asiat voidaan mieltää myös haittana. Looginen, vaiheittainen, muodollinen ja jäykkä rakenne vaikeuttavat myös sen soveltuvuutta kaikkiin hankkeisiin. Joskus organisaatiossa tiedetään mitä ja miksi jotain halutaan mutta välttämättä ei tiedetä kuinka haluttuun loppuratkaisuun päädytään. Voi olla, että vesiputousmalli ei ole paras lähestymistapa hankkeissa, jotka ovat suuria ja huonosti määriteltyjä nopeasti muuttuvassa liiketoimintaympäristössä sekä silloin, kun haluttua lopputulosta ei tunneta hyvin.

Myös merkittävä väärinkäsitys vesiputousmallin luonteesta on, että peräkkäiset vaiheet kehityksen elinkaareissa tarkoittavat sitä, että se varmuudella vähentää riskiä tai toimitusvarmuutta aikataulun ja sisällön suhteen. Vaikka vaiheittain etenevä malli tuntuu täysin loogiselta tavalta lähestyä asioita niin saattaa olla, että virheet ja puutteet lisääntyvät sekä pahenevat peräkkäisessä mallissa ja myöskin epävarmuus lisääntyy, kun lopullinen ratkaisu nähdään vasta prosessin lopussa.

Kun mahdolliset viat paljastuvat esimerkiksi testivaiheessa, jätetään keskeiset sidosryhmät usein raapimaan päätään kysymällä retorisia kysymyksiä: "Miten emme huomanneet sitä?" tai "Miten tämä on mahdollista?". Vaiheittaisella lähestymistavalla koettu hallinta voi ajaa väärin tavoitteisiin. Halutaan vaihe loppuun prosessin mukaisesti, aikataulussa ja sovitussa kustannuksissa välittämättä siitä onko ominaisuudet täydellisesti toteutettu. Tiukka prosessinhallinta rajoittaa ja aiheuttaa muutosten ja uusien ominaisuuksien vastarintaa, koska kaikki toiminnot halutaan toimittaa suunnitellun aikataulun sekä jäädytetyn ominaisuusluettelon ja kustannusarvion rajoissa. Malli asettaa haasteen, kun sitä tarkastellaan organisaation edun kannalta. Organisaation edun kannalta kohdataan tilanteita, joissa olisi parempi, mikäli olisi mahdollista muuttaa projektin laajuutta, aikaa, kustannuksia, ominaisuusluetteloa jne. liiketoimintatarpeiden muuttuessa. Voidaan ajatella, että hanke on todella hyvä prosessilla toteutettavaksi, mutta lopulta valittu prosessi vain tuottaa vääriä tai vaillinaisia ominaisuuksia. (Davis et al. 2014)

Suurin vesiputousmallin haaste on, kuinka se käsittelee muutostenhallinnan projektin aikana. Koska vesiputousmallissa edetään lineaarisesti vaiheittain siinä ei ole mahdollista palata edelliseen vaiheeseen, kun johtoryhmä on hyväksynyt edellisen vaiheen tarkistuspisteessä, vaikka odottamattomia muutoksia ilmaantuisi. Periaatteessa, kun vaihe on

valmis, niin se on siinä. Muutoksia ei voida tehdä helposti. On erittäin vaikeaa ja kallista mennä mallissa taaksepäin, jos esimerkiksi huomataan testausvaiheessa, että jokin ominaisuus on toteuttamatta, kun sitä ei ollut vaatimusmäärittelyssä.

Loppukäyttäjät näkevät toimitettavan ratkaisun vesiputousmallissa myöhäisessä vaiheessa. Projektissa edetään kolme vaihetta ennen kuin toteutus alkaa ja sen jälkeen sidosryhmät näkevät testausvaiheessa toimivan ratkaisun, joka on projektin elinkaaren loppuvaiheessa.

Vaatimusten kerääminen on suhteellisen haasteellista. Malli määrittää, että alkuvaiheessa keskustellaan asiakkaiden ja sidosryhmien kanssa, jotta vaatimukset tunnistettaisiin mahdollisimman hyvin. Saattaa olla haastavaa sidosryhmien kannalta tunnistaa alkuvaiheessa tarkasti mitä he haluavat. Yleensä projektin edetessä he oppivat ja tunnistavat paremmin haluamansa vaatimuksen tai ominaisuuden. (Hughes 2013)

Hoogendoorn (2014) listaa perinteisten projektinhallintamallien luonteenpiirteitä seuraavasti. *Tieto katoaa projektista.* Jokainen vaihe toteutetaan erityisillä tehtävärooleilla. Kun vaiheen lopputulos on valmis, rooli poistetaan projektista. Paljon osaamista menetetään projektin jokaisen vaiheen jälkeen. *Kasvava näkemys projektista.* Jokaisen hankkeen vaiheen aikana tiimi saa lisäinformaatiota projektin laajuudesta, käyttäjävaatimuksista ja valitun teknologian mahdollisuuksista. *Vaatimusten muuttuminen.* Keskimäärin 20-25 prosenttia hankkeen vaatimuksista muuttuu sen ollessa käynnissä. Uusia vaatimuksia löytyy, nykyiset vaatimukset muuttuvat tai vanhentuvat. Koska kaikki vaatimukset ovat yksityiskohtaisesti määritelty aikaisemmassa vaiheessa, osa tästä työstä tulee tarpeettomaksi tai se on suoritettava uudelleen. *Valmis.* Aikaisemman vaiheen lopputulosta ei voi muuttaa myöhemmin, on tärkeää toteuttaa jokainen vaihe täysin valmiiksi erityisesti analyysi- ja suunnitteluvaiheiden aikana. Seuraavassa vaiheessa löydettyjä uusia vaatimuksia käsitellään aiempien vaiheiden puutteina. *Dokumentaatiota liikaa.* *Arvioinnin vaikeus.* Jokainen vaihe sisältää erilaisen työnsisällön, joten on vaikeaa antaa hyviä kokonaisarvioita. *Myöhästymisen riski.* Vesiputousmallissa ratkaisu kehitetään projektin myöhäisessä vaiheessa. Joskus useita vuosia kuluu ennen kuin ensimmäinen toteutus on tehty. Tämä tuo suuria riskejä toteutuksen kanssa. Teknologia voi olla vanhentunut ja ohjelmiston testaus alkaa vasta kun ratkaisu on valmis ja näin ollen virheet löydetään projektin loppuvaiheessa.

3.3 Ketterät menetelmät

3.3.1 Ketterät periaatteet

Vuonna 2001 julkaistiin *Agile Manifesto*, jonka tarkoituksena oli luoda pohja ketterille menetelmille. Julkaisua pidetään ketterän kehityksen perusmenetelmänä, johon pohjautuen on syntynyt projektinhallinnan viitekehyksiä, kuten Scrum.

Manifestin sisältö on seuraava:

Me etsimme parempia keinoja ohjelmistojen kehittämiseen tekemällä sitä itse ja auttamalla siinä muita.

Tässä työssämme olemme päätyneet arvostamaan

Yksilöitä ja vuorovaikutusta enemmän kuin prosesseja ja työkaluja

Toimivaa sovellusta enemmän kuin kokonaisvaltaista dokumentaatiota

Asiakasyhteistyötä enemmän kuin sopimusneuvotteluita

Muutokseen reagoimista enemmän kuin suunnitelman noudattamista.

Vaikka oikeallakin puolella on arvoa,

me arvostamme vasemmalla olevia asioita enemmän. (Agile Alliance 2001)

Lisäksi manifestissa luotiin kaksitoista tärkeää periaatetta, joita tulisi noudattaa ketterissä menetelmissä. Ne ovat (Girvan et al. 2017) mukaan seuraavat.

1. Tärkein tehtävä on toimittaa asiakkaalle varhaisessa vaiheessa ja jatkuvasti arvoa tuottavaa ohjelmistoa.
2. Muutokset vaatimuksiin ovat tervetulleita myös myöhäisessä kehitysvaiheessa. Ketterillä prosesseilla muutokset saadaan muutettua asiakkaan kilpailueduksi.
3. Toimitetaan toimiva ohjelmisto asiakkaalle mahdollisimman usein, muutamasta viikosta muutamaan kuukauteen.
4. Liiketoimintahenkilöiden ja kehittäjien on työskenneltävä yhdessä päivittäin koko projektin ajan.
5. Rakennetaan projektia motivoituneiden henkilöiden varaan. Rakenna heille tukea antava ympäristö ja luota heihin, että he saavat työnsä tehtyä.
6. Tehokkain tapa välittää tietoa on keskustelu kasvotusten.
7. Toimiva ohjelmisto on tärkein mittari prosessin edistymiselle.
8. Ketterät prosessit edistävät kestäväää ohjelmistokehitystä. Kehittäjien ja käyttäjien tulisi pitää kehityksessä tasainen vauhti jatkuvasti.
9. Jatkuva huomio tekniseen laatuun ja hyvään suunnitteluun edistää ketteryyttä.
10. Yksinkertaisuus – keskeistä on minimoida turhan työn määrä.
11. Parhaat arkkitehtuurit, vaatimukset ja suunnitelmat syntyvät itse organisoiduista tiimeistä.
12. Säännöllisin väliajoin tiimin tulee pohtia, miten toimia tehokkaammin, ja sen jälkeen hienosäätää toimintaa sen mukaan.

Cobb (2011) kuvaa kirjassaan ketterien menetelmien suunnittelun tasot seuraavasti.

- Vision suunnittelun tarkoitus on määrittää liiketoiminnan tavoitteet ja päämäärät.
- Tiekartan pyrkimyksenä on rakentaa projektin etenemissuunnitelma ja jakaa visio pienempiin julkaisuihin, jotka toimitetaan tietyn ajan kuluessa. Jokainen jul-

kaisu on yleinen toiminnallisuus, joka toimitetaan inkrementaalisesti loppukäyttäjälle.

- Julkaisun suunnittelun tarkoituksena on jakaa julkaisu iteraatioihin ja näin ollen kunkin julkaisun edellyttämä toiminto kehitetään vähitellen sekä vaiheittain. Jokaisen iteraation tuloksena pitäisi olla toimiva ohjelmisto.
- Iteraation tavoitteena on määrittää tehtävät, jotka on suoritettava kyseisen iteroinnin aikana, jotta se kehittää kyseiselle iteraatiolle tarvittavat käyttäjätarinat.
- Päivittäiset suunnitteluprosessit tarkastelevat ensisijaisesti meneillään olevaa iteraatiota varten suunniteltujen tavoitteiden edistymistä. Tavoitteena on myös tunnistaa sekä ratkaista esteet, jotka voivat hidastaa edistymistä.

Ketterien menetelmien yhteisiä piirteitä ovat iteratiivisuus ja yhteistyö asiakkaan kanssa. Menetelmien eroja ovat niiden keskittyminen ratkaisun elinkaaren eri vaiheisiin ja menetelmän ohjeistuksen tarkkuus. Kun menetelmiä sovelletaan organisaation omiin tarpeisiin, on tärkeää muistaa ketterien menetelmien arvot ja periaatteet.

3.3.2 Ketterien menetelmien vahvuuksia

Layton (2012) listaa kymmenen pääkohtaa ketterien menetelmien eduista projektinjohtamisen kannalta.

Parempi tuotteen laatu

Ketterillä menetelmillä on erinomaiset valmiudet varmistaa, että laatu on mahdollisimman korkea. Ketterät projektitiimit auttavat varmistamaan laadun seuraavilla tavoilla:

- Kehitystiimi toimii proaktiivisesti virheiden välttämiseksi.
- Kehitystiimi hyväksyy teknologisen huippuosaamisen, hyvän suunnittelun ja kestävän kehityksen.
- Kehitystiimi tekee vaatimusmäärittelyn ja kehittämisen juuri ajoissa niin, että tuoteominaisuuksien tuntemus on mahdollisimman tarkoituksenmukainen.
- Julkaisun hyväksyntäkriteerit määritellään käyttäjätarinoihin, jotta kehitystiimi ymmärtää paremmin vaatimukset ja tuoteomistaja voi validoida ne.
- Kehitysprosessi sisältää jatkuvaa integraatiota ja päivittäistä testausta.
- Kehitystiimi käyttää hyväksi automatisoituja testaustyökaluja.
- Sprintin retrospektiivien avulla kehitystiimi parantaa jatkuvasti prosessejaan ja työskentelytapojaan.
- Kehitystiimi seuraa työn valmiutta valmiin määritelmän avulla - kehitetty, testattu, integroitu ja dokumentoitu.

Korkeampi asiakastyytyväisyys

Ketterät tiimit ovat sitoutuneet kehittämään tuotteita, jotka täyttävät asiakkaiden vaatimukset.

- Tiimin tekee yhteistyötä asiakkaan kanssa ja pitää asiakkaan mukana koko hankkeen ajan.

- Tiimissä on tuoteomistaja, joka on asiantuntija tuotevaatimuksista ja asiakkaan tarpeista.
- Tiimi pitää tuotteen työjonon ajan tasalla ja priorisoi työjonoa, jotta voidaan nopeasti vastata muutoksiin.
- Tiimi esittelee toiminnallisuuden asiakkaille kaikissa tarkistuspisteissä.
- Tiimi toimittaa tuotteet markkinoille nopeammin ja useammin jokaisen julkaisun myötä.

Korkeampi tiimin moraali

Työskentely onnellisten ihmisten kanssa, jotka nauttivat työstään - tyydyttää ja palkitsee. Ketterä projektinhallinta parantaa tiimin moraalia seuraavilla tavoilla:

- Olemalla osa itseohjautuvaa tiimiä mahdollistetaan luovuus, innovatiivisuus ja tunnustetaan asiantuntemus.
- Keskittyminen kestäviin työtapoihin, joka takaa, että ihmiset eivät polta itseään loppuun stressiin tai ylitöihin.
- Poistetaan esteet ja suojataan kehitystiimiä ulkopuolisilta häiriöiltä.
- Tuen ja luottamuksen ilmapiiri lisää ihmisten yleistä motivaatiota ja moraalia.
- Kasvotusten tapahtuva kommunikaatio auttaa vähentämään harhaanjohtavan viestin aiheuttamaa turhautumista.
- Työskentelytavat auttavat kehitystiimin jäseniä oppimaan uusia taitoja ja kasvamaan opettamalla toisiaan.

Lisää yhteistyötä ja omistajuutta

Kun kehitysryhmät ottavat vastuun projekteista ja tuotteista, ne tuottavat hyviä tuloksia. Ketterät kehitystiimit toimivat yhteistyössä ja omaksuvat tuotteiden laadun ja hankkeen suorituskyvyn omistukseensa seuraavasti:

- Kehitystiimi, tuoteomistaja ja projektipäällikkö työskentelevät tiiviisti yhdessä päivittäin.
- Järjestetään sprintin suunnittelukokouksia, joiden avulla kehitystiimi voi järjestää työnsä.
- Organisoidaan kehitystiimin johtama päivittäinen kokous, jossa kehitystiimin jäsenet kommunikoivat tehdyn työn, tulevan työn ja mahdolliset esteet.
- Kehitystiimi esittelee katselmuksissa tehdyt työt ja kommunikoi tuotteesta suoraan sidosryhmien kanssa.
- Suoritetaan sprintin retrospektiivi, jonka avulla kehitystiimin jäsenet voivat tarkastella työskentelytapoja ja ottaa käyttöön parempia käytäntöjä seuraaviin iteraatioihin.
- Työskennellään ympäristössä, joka mahdollistaa nopean viestinnän ja yhteistyön kehitystiimin jäsenten kesken.
- Tiimi tekee päätöksiä yksimielisesti.

Räätälöidyt tiimit

Ajatus tiimien räätälöinnistä mahdollistaa ketterien työpaikkojen lisääntyvän monimuotoisuuden. Organisaatiot, joilla on perinteiset johtamistyyli, ovat yleensä monoliittisia tiimejä, joissa jokainen noudattaa samoja sääntöjä. Ketterillä hankkeilla on ihmisiä, joilla on hyvin erilaisia vahvuuksia ja joiden vahvuudet auttavat tekemään hyviä tuotteita, kun ne on tunnistettu.

Asiaankuuluvat mittarit

Mittarit, joita ketterät projektitiimit käyttävät arvioimaan aikaa ja kustannuksia sekä mittaamaan projektin suorituskykyä ja tekemään hankkeiden päätöksiä, ovat usein tärkeämpiä ja tarkempia kuin perinteisten projektien tiedot. Ketterissä projekteissa tuotetaan seuraavia mittareita.

- Hankkeen aikataulut ja budjetit määritetään jokaisen kehitystiimin todellisen suorituskyvyn ja kykyjen perusteella.
- Kehitystiimit antavat ja vastaavat hankkeen työmääräarvioista.
- Yksittäisen kehitystiimin kyvykkyys ja ominaisuudet määrittävät arvioidun työmäärän.
- Kun kehitystiimi oppii lisää tehtävästä projektista, niin arviointi paranee koskien työtehtäviä, aikaa ja kustannuksia.
- Päivittäin päivitetty iteraation Burn-Down-kaavio antaa tarkat tiedot siitä, missä vaiheessa kehitystyö on menossa.
- Kehityskustannusten vertaaminen tulevan kehityksen kustannuksiin tarkentuu, kun osaamispääoma siirretään uudelle hankkeelle, joka auttaa projektitiimejä määrittämään tarkemmin projektin kestoa.

Näkyvyyden parantuminen

Ketterillä projekteilla jokainen projektiryhmän jäsen tietää, miten hanke etenee. Ketterät projektit tarjoavat korkeantason näkyvyyden projektinjohdolle.

- Projektin viestintä on avointa ja rehellistä sidosryhmien, asiakkaiden ja muun organisaation sisällä.
- Projekti tarjoaa päivittäiset mittaukset sprintin suorituskyvystä sprintin työjonon päivityksillä. Sprintin työjonot ovat käytettävissä organisaatiossa kenelle tahansa, jotta ne voidaan katselmoida.
- Projekti tarjoaa päivittaisen edistymisen seurannan kehitystiimin edistymisestä ja mahdollisista esteistä päivittaisen kokouksen kautta.
- Projekti esittelee saavutukset tarkistuspisteissä. Jokainen organisaation sisällä voi osallistua.

Lisääntynyt projektin kontrolli

Ketterillä projektitiimeillä on lukuisia mahdollisuuksia hallita projektin suorituskykyä ja tehdä korjauksia tarpeen mukaan seuraavista syistä:

- Painopisteiden muuttaminen koko projektin ajan mahdollistaa hankkeiden sopeuttamisen liiketoiminnan muutokseen.
- Muutoksien hyväksyminen sallii projektiryhmän reagoida ulkopuolisiin tekijöihin, kuten markkinoiden kysyntään.
- Päivittäiset kokoukset mahdollistavat tiimin nopean ongelmien ratkaisemisen niiden syntyessä.
- Jokapäiväiset päivitykset auttavat tarkastelemaan sprintin suorituskykyä, jolloin voidaan reagoida ongelmiin.
- Kasvotusten tapahtuva kommunikointi parantavat viestintää.
- Sidosryhmät näkevät hankkeen edistymisen tarkistuspisteissä.
- Sprintin retrospektiivit mahdollistavat tiimin jatkuvan kehittämisen tuotteiden laadun parantamiseksi, kehitystyön suorituskyvyn lisäämiseksi ja projektiprosessien parantamiseksi. Kaikilla mahdollisuus osallistua seuraamaan tapahtumaa.

Lisääntynyt ennustettavuus

Ketterät projektinhallintamenetelmät auttavat projektiryhmää tarkasti ennakoimaan, miten asiat sujuvat hankkeen edetessä. Seuraavassa on joitain käytäntöjä, artefakteja ja työkaluja parantamaan ennustettavuutta:

- Iteraatioiden kestojen yhdenmukaisuus ja kehitystiimien tilannekatsaukset koko projektin ajan mahdollistavat projektiryhmän tunnistamaan tarkat kustannukset.
- Yksittäisen kehitystiimin tunnistetun kehitysnopeuden ansiosta projektiryhmä voi ennakoida julkaisujen aikataulut ja budjetit.
- Päivittäisten kokouksien ja työjonojen avulla projektiryhmä voi ennustaa yksittäisten iteraatioiden suoritustasoa.

Vähentyneet riskit

Ketterät menetelmät eliminoivat lähes kokonaan projektin epäonnistumisen mahdollisuudet joiden seurauksena voidaan kuluttaa suuria määriä aikaa ja rahaa ilman tuottoa. Alla mainitut asiat suoritetaan säännöllisesti ja riittävän lyhyissä aikajaksoissa, jolloin projektin suorituskykyyn voidaan reagoida nopeasti.

- Vaatimukset työstetään jokaiseen sprinttiin niin, että saadaan käyttökelpoisia ominaisuuksia riippumatta siitä, mitä ympäristössä voi tapahtua tulevaisuudessa.
- Kerätään tuotteesta ja prosesseista jatkuvasti palautetta.
- Päivittäiset kokoukset ja jatkuva tiimin kehittäminen sekä viestintä sidosryhmille vähentävät epätietoisuutta.
- Säännöllinen työlistan päivitys uusista ja olemassa olevista vaatimuksista sekä organisaation tuoteomistajan työlistan tarkastelu ja hyväksyntä.
- Sprinteissä suoritetaan katselmukset sidosryhmien ja asiakkaiden kanssa vaatimusten suhteen.
- Kehitystiimi käsittelee prosessin parantamista sprintin retrospektiivissä.
- Julkaisuissa loppukäyttäjä voi nähdä ja reagoida säännöllisesti uusiin ominaisuuksiin ja arvioida ominaisuuden oikeellisuus tarpeen näkökulmasta.

- Arvontuotto varhaisessa vaiheessa, koska julkaisusykliä ovat tiheitä ja tärkeimpiä ominaisuuksia ei jouduta odottamaan hankkeen päätökseen saakka. Katselmuksissa varmistetaan ominaisuuksien toimivuus loppukäyttäjän toimesta.

3.3.3 Scrum viitekehys

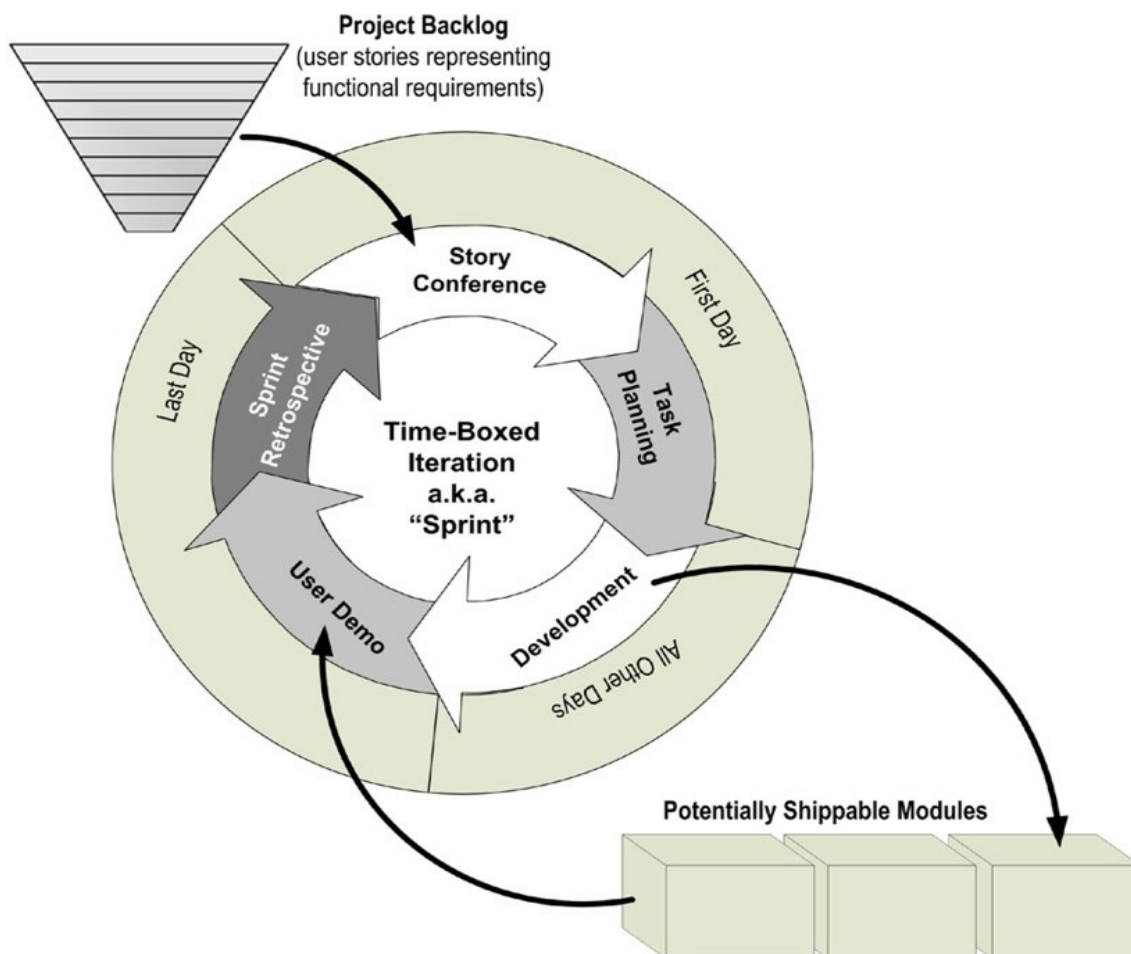
Scrum on iteratiivinen ja ketterä toimintatapa, jossa itseohjautuva tiimi työskentelee yhteisten tavoitteiden eteen. Perinteisessä tuotekehityksessä jaamme hankkeen joukoksi vaiheita ja suoritamme määrättyt tehtävät vaiheittain. Tavoitteena on täyttää asiakkaiden kysyntä tai tarve. Kun tunnistamme asiakkaiden tarpeet, kehitys kulkee läpi eri prosessin vaiheet, kunnes saavutaan lopputestausvaiheeseen prosessissa, jonka jälkeen tuote käyteenotetaan ja toimitetaan asiakkaalle. (Pries et al. 2011)

Scrumissa projektiryhmä tuottaa useita pienempiä osakokonaisuuksia, jotka vähitellen kehittävät tuotetta. Jokainen lisäys testataan ja toimitetaan asiakkaalle sekä saadaan palaute asiakkaalta.

Scrum-periaate on, että usein toistuva tuotejulkaisu sallii muutokset, jotka syntyvät tuotekehityksen aikana. Sen sijaan, että yritetään sisällyttää muutos tuotteeseen, Scrum ja ketterän kehityksen prosessit tukevat mallia, että tuote sopeutuu asiakkaan vaatimuksiin nopeassa syklissä. (Pries et al. 2011)

Asiakkaan vaatimukset ajavat Scrum-prosessia kuten (Hughes 2016) kuva 10 esittää. Tyypillisesti vaatimuslistaa on kuvattu käyttäjätarinoiden listaksi. Käyttäjätarinat ovat yhden lauseen mittaisia tarpeita liiketoiminnan näkökulmasta. Tuotteenomistaja pitää listan vaatimusten mukaisessa tärkeysjärjestyksessä. Tämän jälkeen Scrum-tiimi katsoo, kuinka monta käyttäjätarinaa mahtuu tietyn pituiseen kehityssykliin. Tämän jälkeen toimitetaan asiakkaalle toimiva tuotekokonaisuus tai osakokonaisuus. Sprintin pituus on määritelty kestoltaan pisimmillään yhteen kalenterikuukauteen. Mikäli sprintin kestoa kasvatetaan, suunnittelupalaverissa sovitun sisällön määritelmät saattavat muuttua ja se vaarantaa sprintin onnistumisen. Sprinttien ennustettavuus lisääntyy mahdollistamalla työn tilan tarkastelun ja muutokset kohti sprintin tavoitetta vähintään kerran kuukaudessa. Samalla sprinttien kustannusriski myös rajoittuu enintään yhden kalenterikuukauden kustannukseen. Alussa kehitystiimi yleensä suunnittelee ominaisuudet sekä listatut työaktiviteetit, jotka mahdollistavat toimivan ratkaisuversion kehittämisen. Kehitystiimi suunnittelee sprintin suunnittelupalaverissa sen verran työtä, jonka se uskoo voivansa valmistaa alkavassa sprintissä. Sprintin alkuvaiheeseen suunnitellut työaktiviteetit paloitellaan enintään yhden päivän kokoisiin yksittäisiin tehtäviin palaverin loppuun mennessä. Kehitystiimi itseohjautuu sprintin suunnittelupalaverissa ja sprintin aikana toteuttaa sprintin kehitysjonossa olevat työtehtävät. Kun ominaisuus saadaan valmiiksi, se esitellään asiakkaalle demokatselmuksessa. Demokatselmuksissa asiakas kykenee arvioimaan, onko ominaisuudet niitä, joita on pyydetty. Ennen seuraavaa sprinttiä, kehi-

tystiimi järjestää sprintin retrospektiivin, jossa he keskustelevat prosessien ja toimintatapojen hyvistä ja huonoista puolista, joita tiimi on kohdannut kehitysvaiheen aikana.



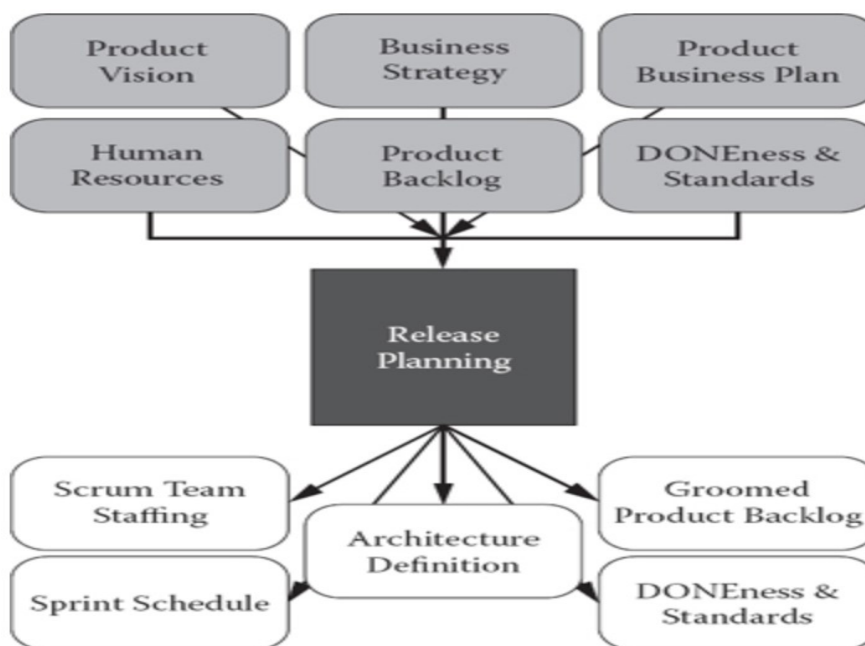
Kuva 10. Scrumin kehitysiteraation rakenne ja vaiheiden kesto

Hughes (2016) esittelee kuvassa 10 Scrumin iteraation viidessä vaiheessa tapahtuvat toimitukset. Näitä ovat sprintin suunnittelupalaveri, tehtävien suunnittelu, kehitys, käyttäjädemo, sprintin retrospektiivi. Kuvassa 10 ulkokehällä näytetään vaiheiden ajallinen riippuvuus koko sprintin keston. Kehitys- ja demovaiheen jälkeen projekti tuottaa julkaistavan osan järjestelmästä. Tämän jälkeen, tiimi on valmis aloittamaan toisen syklin. Kehitystyö jatkuu niin kauan kuin asiakkaalla riittää käyttäjätarinoita tai sponsorit jatkavat rahoitusta. (Hughes 2013)

Sprintin kesto on yleisimmillään neljä viikkoa mutta erittäin poikkeuksellisissa olosuhteissa tehtäviä voi lisätä tai poistaa sprintistä, jos tuoteomistaja ja tiimi hyväksyvät muutokset. Muutokset sprintin aikana ovat poikkeustapauksia - ei käytössä oleva normi. (Pham et al. 2011)

Ennen projektin alkua kannattaa viettää aikaa valmistautumisessa projektiin. Projektin sisällön suunnittelu voidaan toteuttaa yhdessä tai kahdessa työpajassa. Vaihtoehtona on, että käynnistetään niin sanottu valmisteluiteraatio projektin suunnittelua varten ja näin ollen voidaan varmistaa, että projekti tuottaa halutun lopputuloksen.

Alla olevassa kuvassa 11 on esitetty tehtävävaiheita, joita Schiel (2013) pitää huomioitettavina valmistelevina vaiheina uuden julkaisun suunnittelussa.



Kuva 11. *Julkaisun suunnittelukokouksen syötteet ja tulokset*

- Valmistellaan hankkeen tai julkaisun työlista, määritetään ja valitaan vaatimukset.
- Asetetaan Sprintin aikataulu.
- Asetetaan Scrum-tiimien henkilöt.
- Järjestetään Scrum-tiimien koulutus.
- Tarkastellaan ja muutetaan tarvittaessa valmiin määritelmä.
- Luodaan tai muutetaan arkkitehtuurin määritelmä.
- Luodaan työlista ensimmäiselle Sprintille.

Ennen sprintin suunnittelukokousta projektitiimillä on tiedossa tuotteen visio, yrityksen liiketoimintastrategia, tuotteen liiketoimintasuunnitelma, käytettävissä oleva henkilöstö, tuotteen työlista ja valmiin määritelmä sekä standardit. Julkaisun suunnittelukokouksessa määritellään ja sovitaan seuraavissa vaiheissa tarvittava henkilöstö osaamistarpeiden ja kyvykkyyden mukaan sekä tarvittaessa järjestetään koulutusta osaamisen vahvistamiseen. Suunnittelukokouksessa asetetaan myös sprintin aikataulu, sovitaan työlistan sisältö, luodaan ratkaisun arkkitehtuurin määritelmä ja sovitaan valmiin määritelmät ominaisuuksille.

3.4 Scaled Agile Framework – SAFe

Knaster (2017) määrittelee mallin seuraavasti. SAFe-malli on menetelmä koko yrityksen ketteryteen ja se muokataan kunkin yrityksen vaatimusten mukaiseksi. Mallissa toimiminen edellyttää kokonaisvaltaista ketterää toimintaa kaikilla yrityksen osaluilla, ei vain ratkaisun kehityksessä.

SAFe-mallin mukainen ketterä lähestymistapa on käytössä kansainvälisesti eri toimialoilla. Mallin lähtökohtana on Lean-periaatteet: Jatkuva parantaminen (Kaizen) ja hukan eliminointi.

SAFe-malli kattaa yrityksen toiminnot hankkeen toteutustasosta operatiiviseen portfolioon johtamiseen, investointimenetelmiin, tulevaisuuden suunnitteluun ja toteutuneiden seurantaan. Laaja-alainen ketteryys mahdollistaa mukautuvan johtamisen kaikilla organisaation tasoilla.

SAFe-mallin hyödyt organisaatiolle ovat seuraavat. Malli helpottaa hankkeiden läpivientiä, sisällön priorisointia sekä riippuvuuksien hallintaa. Hankeissa tehdään asiakkaan kannalta oikeita asioita oikeassa aikataulussa ja hyötyihin sisältyvät myös portfolioon priorisointi, päätöksenteon selkeys ja resurssiensuunnittelun johdonmukaisuus. (Lefingwell 2016)

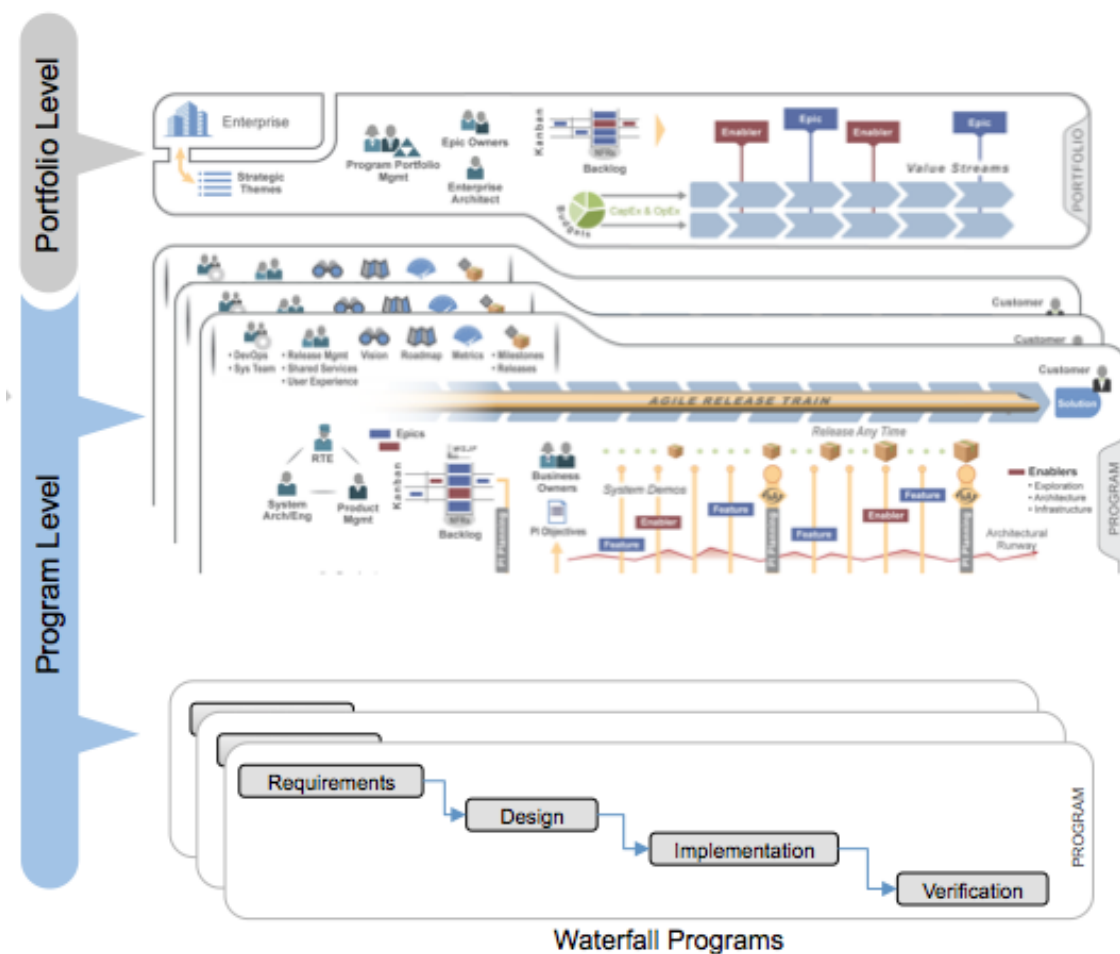
SAFe-mallin ohjaavia periaatteita ja arvoja ovat.

- Aikainen arvontuotto asiakkaalle. Asiat pyritään viemään käytäntöön ja asiakas-hyötyyn mahdollisimman pian.
- Pieni eräkokoo, jolla mahdollistetaan nopea läpimenoaika.
- Läpinäkyvyys koko hankkeen ajan.
- Kehittäminen on syklistä.
- Epävarmuuden sietäminen.
- Jatkuva parantaminen, hukan eliminointi ja kehittämismallin empiirinen luonne.

Salkkutaso - portfolio

Salkkutaso organisoii ketterässä yritys ympäristössä organisaation läpi kulkevia arvovirtoja, joiden hallintaan tarvitaan järjestelmiä tavoitteiden saavuttamiseksi. Salkkutaso sisältää strategiset teemat, jotka ovat tunnistettuja liiketoiminnan tavoitteita määriteltynä yrityksen visiossa. Salkkutasoon sisältyy myös yrityksen arvovirrat, jotka ovat vastuussa ratkaisujen ja palveluiden toimittamisesta, jotta yritys saavuttaa tavoitteensa. Taso sisältää myös salkunhallinnan. Salkunhallinta (PPM) edustaa tahoja, joka omistaa yrityksen tuote- tai palvelusalkun korkeimman tason strategisen ja lainvoimaisen päätäntävalan. PPM:llä toimintona on vastuu strategiasta ja investointien rahoittamisesta, hankkeiden hallinnasta sekä johtamisesta. (Knaster 2017)

Salkunhallinnassa, joka on esitetty kuvassa 12, hyödynnetään vesiputousmallin ja ketterän kehittämisen menetelmiä (Moran 2015). Yhteisen kehitysrytmin löytäminen, joka synkronoi kaikki hankkeet ja kehitystiimit sekä sidosryhmät, on keskeinen mekanismi, jonka avulla jatkuva arvontuotto voidaan saavuttaa. Organisaatioiden eri toimintojen on usein vaikea siirtyä samalla aikataululla ketteriin menetelmiin eri syistä. Näitä syitä voivat olla riippuvuudet ulkoisiin kumppaneihin ja organisaatioihin, jotka saattavat käyttää vesiputous-menetelmää sekä sisäinen kulttuuri, joka vaatii aikaa siirtymisessä ketterään ajatteluun.



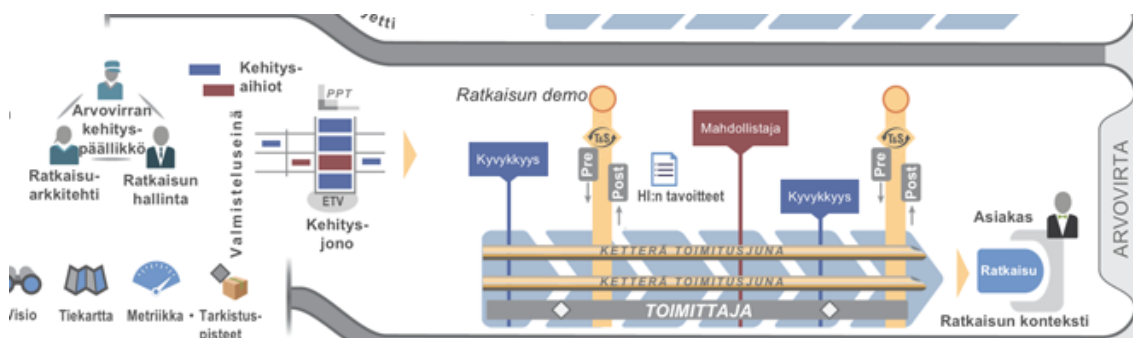
Kuva 12. Hankkeen salkunhallinnassa hyödynnetään ketterää ja vesiputousmenetelmää

Salkkutason tavoitteena on varmistaa, että viedään toteutukseen kerralla vain sen verran kehitysaihia, jotka kyetään viemään läpi tuote- ja palvelukehityksen kapasiteetilla ja kyvykkyydellä. Portfoliossa tulee myös tunnistaa arvovirroilta haluttavat liiketoimintahyödyt tuote- tai palvelukohtaisten vaatimusten sijaan. Arvovirtaa johdetaan tiekartan avulla. Tiekartassa suunnitellaan hanketta eteenpäin pitkällä aikavälillä ja hankkeen lähiaika määritellään tarkemmin sekä hyväksytään, että tulevaisuus tarkentuu ajan kuluessa. Tässä vaiheessa myös pilkotaan ja vaiheistetaan tulevat tehtävät, jotta saavutetaan pienen eräköön vaatimukset.

Vaiheessa tulee myös tunnistaa arkkitehtuuriratkaisut, jotka tulee toteuttaa osana investointia. Portfoliotason tehtävät ja saavutettu lopputulos määräytyvät vaiheessa muodostettavan salkun kehitysjonon mukaan. Siihen hyväksytyt kehitysaihiot – liiketoiminnan ja arkkitehtuurin (Measey 2015) – järjestetään jonoon toteutusjärjestyksessä eli mitkä otetaan työn alle ja mitkä suunnitellaan käynnistettävän vasta myöhemmin. Työn alle otetaan tehtäviä käytettävissä olevan kapasiteetin mukaan. Arvovirran uusia kehitystarpeita ja vanhojen vaatimusten jatko- ja pienkehitystä tarkastellaan kokonaisuutena. Kehitysaihion valmistelussa analysoidaan tavoitteet ja karsitaan ominaisuudet, jotka eivät tuota lisäarvoa. Valmistelussa myös tunnistetaan minimitoiminnallisuus, joka mahdollistaa aikaisen arvontuoton. Valmisteluvaiheen avulla tunnistetaan liiketoiminnallisten vaatimusten lisäksi arkkitehtuuriset ratkaisut, jotka ovat edellytys ratkaisun toteutukselle. Hankkeen kehityssuunnittelua voidaan tehdä esimerkiksi koko vuoden ajalle vesiputousmallilla ja ketterillä menetelmillä toteutetaan lyhyemmän aikajakson kehitysvaiheet ketterissä toimitusjunissa. Ketterillä toimitusjunilla varmistetaan eräkoon pienentäminen ja riittävän nopea läpimenoaika, koska toimitusjuna sisältää ainoastaan sen verran kehitysaihioita, jotka voidaan sovitussa ajassa toimittaa. Toimitusjuna on pysyväisluontoinen useampien ketterien tiimien muodostama tiimi. Toimitusjuna hitsaa tiimit yhteen yhteiseen tehtävään sekä tarjoaa säännöllisen rytmin suunnittelulle, kehitykselle ja retrospektiiville. Salkunhallinta antaa myös mahdollisuuden tunnistaa arkkitehtuurin edellytyksiä. (Leffingwell 2016)

Arvovirta – value stream

Kehityksen arvovirta on joukko määrittely-, kehitys-, testaus- ja käyttöönoton prosessien tehtäviä, joiden avulla toteutetaan arvoa liiketoiminnalle, asiakkaalle ja loppukäyttäjälle.



Kuva 13. Arvovirta

Kuvassa 13 esitettyä arvovirtaa toteutetaan *Leanin* ja ketterien periaatteiden mukaisesti, jatkuvan kehityksen toimitusjunissa. Ketterä toimitusjuna tuottaa ratkaisun, joka auttaa yritystä ratkaisemaan kohtaamiaan haasteita. Arvovirta voidaan kuvata syklinä – tunnistaa tarpeet, toteuttaa ratkaisu ja julkaise. (Knaster 2017)

Hanketaso - program

Hanketason tavoitteena, joka esitetään kuvassa 14, on ohjata toteutusta sisällön ja tavoitteiden kautta huomioiden elinkaarikustannukset sekä luoda läpinäkyvyys tekemiseen ja syntyviin tuloksiin niin toimitusjunan sisällä kuin sen ulkopuolellakin. Hanketason tavoitteena on saada aikaan jatkuvan kehittämisen virta, jossa minimoidaan toimitusjunan tai hankkeen käynnistys- ja lopetusvaiheen kustannukset. Riskien minimointi kokeilemalla, testaamalla ja vuorovaikutuksen kautta saatavalla palautteella nopeissa sykleissä - ominaisuuksia ei tehdä varastoon.

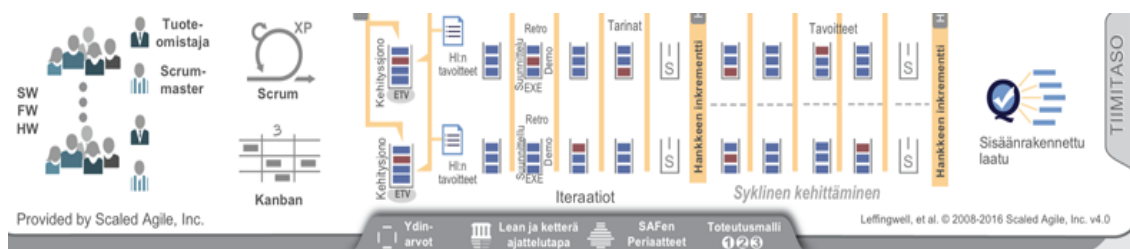


Kuva 14. Hanketaso

Vaiheessa rakennetaan hankkeen kehitysjono pilkkomalla portfoliotasolla valmistellut tavoitteet pienempiin osiin eli ominaisuuksiin ja näille suunnitellaan alustava julkaisu-aikataulut ominaisuuksittain. Hankkeen kehitysjonosta otetaan ominaisuuksia tarkennettavaksi tekemisyjärjestyksessä ja kuvataan ominaisuudet sille tasolle, että toteutustiimi ymmärtää mitä ominaisuudelta vaaditaan. Tarkentaminen on jatkuva prosessi ja sitä tehdään sitä mukaan, kun toteuttajat saavat ominaisuuksia valmiiksi ja vapautuu kapasiteettia uusille kehitysaioille. Vaiheessa myöskin suunnitellaan ja toteutetaan julkaisujen lanseeraaminen ja käyttöönotto koulutuksen, sisällön ja palvelujen osalta. Tärkeää on myöskin hallita riippuvuudet muihin hankkeisiin ja vuorovaikutus eri sidosryhmien kanssa. Loppuvaiheessa varmistetaan sidosryhmiltä, että toteutettava ratkaisu täyttää asetetut tavoitteet sekä toteuttaa tiekarttaa, että visiota ratkaisusta. (Leffingwell 2016)

Tiimitaso

Tiimitason vaiheet on esitetty kuvassa 15. Tiimitason tavoitteena on varmistaa, että toteutustiimillä on edellytykset suunnitella, toteuttaa ja testata ratkaisu. Tiimi saa tarvitsemansa tuen hanketasolta. Tiimin tavoitteena on kyetä tuottamaan ja esittelemään saatuja tuloksia tuote- ja ratkaisunhallinnalle. Tärkein mittari työlle on toimiva sovellus ja valmius tuottaa uusissa inkrementeissä jatkuvasti toimivaa ratkaisua.



Kuva 15. Tiimitaso

Tiimit ovat itseohjautuvia ja priorisoivat, suunnittelevat ja toteuttavat ominaisuuksia hanketason kuvaamien vaatimusten mukaan. Ketterä tiimi toimii nopeissa sykleissä. Tiimin kehitysjonossa vaatimukset kuvataan käyttäjätarinoina ja päätökset sekä työmäärän estimoinnit tehdään yksityiskohtaisesti. Laatu on tiimin yksi tärkeä arvo ja laatu tuotetaan automatisoimalla testausta, versionhallintaa ja jakelua kaikilla tasoilla. Nopean toimitusajan takaa pieni erä koko ja valmista tehdään ominaisuus kerrallaan. Hanketasolle vuorovaikutus tapahtuu tuoteomistajan ja Scrum Masterin kautta. Toteutustiimien toiminta on synkronoitu, iteraatiot alkavat ja päättyvät samaan aikaan, jokaisen iteraation päätteeksi syntyneet tuotokset integroidaan ja testataan kokonaisuutena - näin voidaan paremmin hallita myös riippuvuuksia. (Leffingwell 2016)

3.5 Disciplined Agile Delivery - DAD

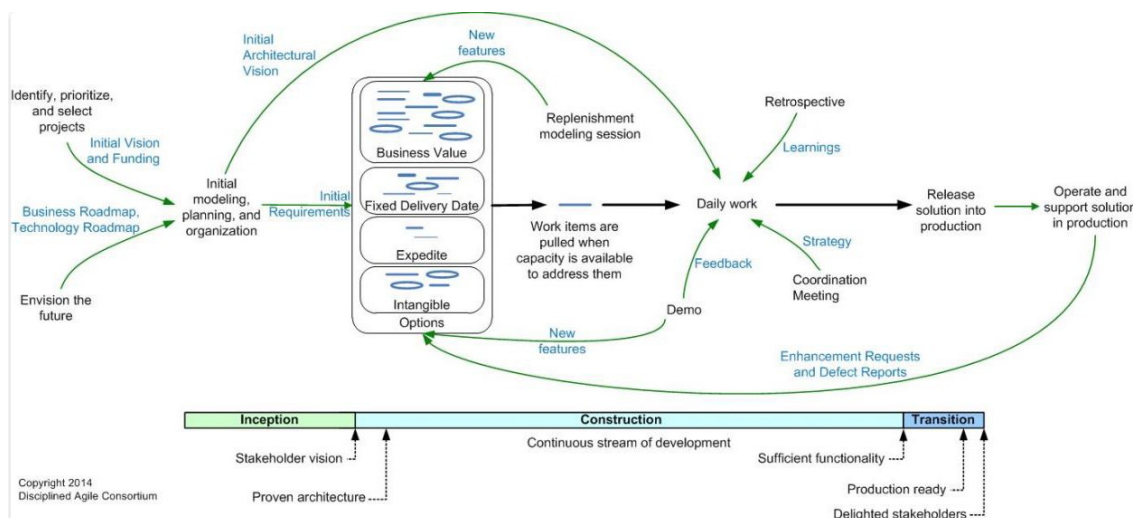
Useat yritykset käynnistävät ketterän kehityksen ottamalla käyttöön Scrum-menetelmän, koska se on toimiva menetelmä ketterien tiimien toimintaan sekä johtamiseen. Scrum on kuitenkin vain osa sitä, mitä vaaditaan, kun toimitetaan kehittyneitä ratkaisuja asiakkaalle. Tiimit joutuvat täydentämään muista menetelmistä prosessiin jääviä puutteita, jotka Scrum on tarkoituksella jättänyt huomioimatta. Tarkasteltaessa muita menetelmiä on huomattava päällekkäisyys ja ristiriidassa oleva terminologia, joka voi olla hämmentävää ammattilaisille ja ulkopuolisille sidosryhmille. (Ambler 2014)

Näiden haasteiden ratkaisemiseksi Disciplined Agile Delivery (DAD) päätöksentekoprosessi tarjoaa johdonmukaisemman lähestymistavan ketterän ratkaisun toimitukseen. Tarkemmin määritelmä on: "Disciplined Agile Delivery (DAD) menetelmän raami, joka tarkoittaa käsitteitä hybridi, ketterä, riski- ja arvosuuntautunut, toimituskeskittynyt, IT-ratkaisukeskittynyt, skaalautuva, yritystietoinen, päämääräsuuntautunut, ihmiset ensimmäiseksi ja oppimissuuntautunut". (Ambler 2014)

3.5.1 DAD-menetelmä ja päämäärät

DAD-menetelmässä on selvästi mielenkiintoisia näkökohtia. DAD on hybridi-lähestymistapa, mikä laajentaa Scrum-menetelmää. DAD on ei-vapaamuotoinen, vapaasti käytettävissä oleva kehys. DAD laajentaa Scrumin toteutukseen keskittynyttä elinkaarta niin, että näkökulmat projektin alustamisesta ratkaisun toimittamiseen loppu-

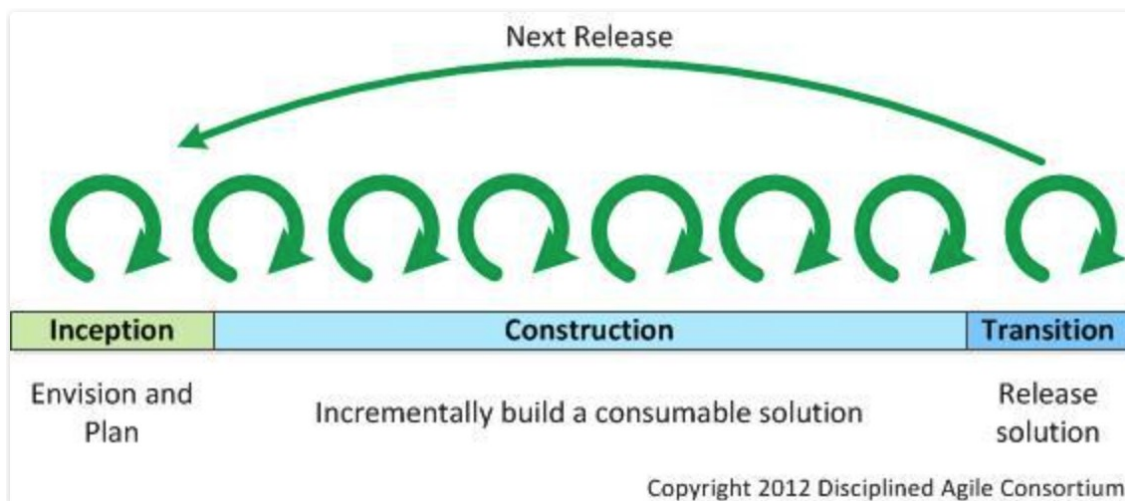
käyttäjille on menetelmässä otettu huomioon. Se tukee myös leanin ja jatkuvan toimituksen elinkaarta toisin kuin muut ketterät menetelmät. DAD ei määrittele yhtä elinkaarta, koska se tunnustaa, että yksi prosessikoko ei sovi kaikkiin hankkeisiin. DAD sisältää ohjeita ja teknisiä käytäntöjä Extreme Programming -menetelmässä (XP) käytetyistä teknisistä käytännöistä sekä mallintamis-, dokumentointi- ja hallintastrategioista, jotka puuttuvat Scrumista ja XP:stä. DAD-kehys ottaa tavoitteisiin perustuvan lähestymistavan. Näin DAD tarjoaa kontekstuaalisia ohjeita liittyen toimivista vaihtoehtoista ja niiden kompromisseista, joiden avulla voidaan räätälöidä DAD:n tehokkaasti ratkaisemaan haasteet. Menetelmä kuvaa mikä toimii, mikä ei toimi ja tarjoaa strategioita, joita voidaan ottaa käyttöön tarvittaessa. (Ambler 2014)



Kuva 16. DAD advanced/lean elinkaari

DAD tarjoaa useita elinkaarimalleja, joita ovat Agile/Basic laajennettuna Scrumilla, Advanced/Lean, jatkuva toimitus ja lean startup. Kuvassa 16 on esitelty (Ambler 2014) advanced/lean elinkaarimalli, jonka luonteenomaisia piirteitä ovat tuki jatkuvalla toimitukselle, jonka tarkoituksena on toimittaa niin usein kuin mahdollista. Aina kun kapasiteettia tiimillä vapautuu, niin uusi kehitysaihio annetaan tiimille. Käytännöt ovat myös tiimin itsensä määrittelemiä kuten suunnittelu, retrospektiivi, mallinnus jne. Tiimillä on oma kehitysaihiolistansa, jossa aihioita ei käsitellä samanarvoisina. Kehitysaihiot voidaan jakaa eri kategorioihin arvontuoton, sovitun toimituspäivämäärän, kriittisyyden tai muun luokittelun mukaisesti. Kuvan 16 mallissa on alustus-, rakennus- ja siirtymisvaihe. Elinkaari soveltuu erinomaisesti kokeneille tiimeille, jotka julkaisevat toiminnallisuksia säännöllisesti ja näin tuetaan jatkuvaa toimitusstrategiaa Lean-lähestymistavalla. Kehitysvaiheen eri toimintoja suoritetaan jatkuvasti tarpeen mukaan koko rakennusvaiheen ajan, kuten katselmuksia, retrospektiivejä, uusien vaatimuspyyntöjen käsittelyä. Ratkaisun toteutustyö on etusijalla ja vaatimukset tuodaan kehitystiimille juuri silloin, kun toiminnallisuus tarvitaan loppukäyttäjille ja näin ollen keskenraisten aihioden määrä on minimoitu.

Yksi Disciplined Agile (DA) -prosessin päätöksentekokehyksen avainkysymyksistä on se, että se edistää kokonaisvaltaista, alusta loppuun sopivaa ratkaisun toimitusjaksoa. Kuvassa 17 tarkastellaan korkean tason näkymää DA-toimituksen elinkaaresta. Se on kolmivaiheinen elinkaari, jossa kehitetään vaiheittain ratkaisua toimitukseen. (Ambler 2014) Elinkaaren päävaiheet ovat alustus-, rakennus- ja siirtymisvaihe. Prosessissa on myös koko ajan meneillään olevia huomiota vaativia asioita.



Kuva 17. DA elinkaaren vaiheet

Alustusvaihe. Alustusvaiheen aikana, joka esitetään kuvassa 18, alustetaan projektin aloittamisvaihe. Tosiasia on, että valtaosa tiimeistä tekee jonkin verran etukäteistyötä projektin alussa. Alustusvaiheessa voidaan käynnistää valmisteluiteraatio, joka keskimäärin kestää kauemmin kuin yksi rakennusvaiheen iteraatio (vuoden 2013 Agile Project Initiation-tutkimuksessa keskimääräinen ketterä tiimi kuluttaa noin kuukauden alustusvaiheeseen, kun iteraation pituus on kaksi viikkoa). DAD:n aloitusvaiheessa tehdään joitakin erittäin kevyitä visiointitehtäviä projektin kehykseen. Projektioorganisaatiolta vaaditaan kurinalaisuutta pitää alustusvaihe riittävän lyhyenä. Alustusvaiheen tehtäviä ovat muodostaa alustava tiimi, määrittää yhteinen visio, noudattaa yrityksen toimintalinjoja, tutkia alustava raami kehitykselle, identifioida alustava tekninen strategia, kehittää alustava julkaisusuunnitelma, pystyttää työympäristöt, turvata rahoitus ja tunnistaa risikit. (Ambler 2014)



Kuva 18. Alustusvaiheen tavoitteet

Rakennusvaihe. Tässä vaiheessa kehitystiimi tuottaa toimivan ratkaisun inkrementaalisesti. Rakennusvaiheessa voidaan tehdä useita iteraatioita tai tehdään jatkuvan kehityksen mukaisesti. Tiimi soveltaa Scrumin, XP:n, Agile Modelingin, Agile Data -ratkaisujen ja muiden menetelmien hybridiä ratkaisun toimittamiseen. Rakennusvaiheen tavoitteet on esitetty kuvassa 19. Tavoitteet ovat käytettävän ratkaisun tuottaminen, sidosryhmien muutostarpeiden käsittely, siirtyminen kohti käyttöönotettavampaa versiota, laadun parantaminen ja varhain ymmärretty arkkitehtuuri. (Ambler 2014)



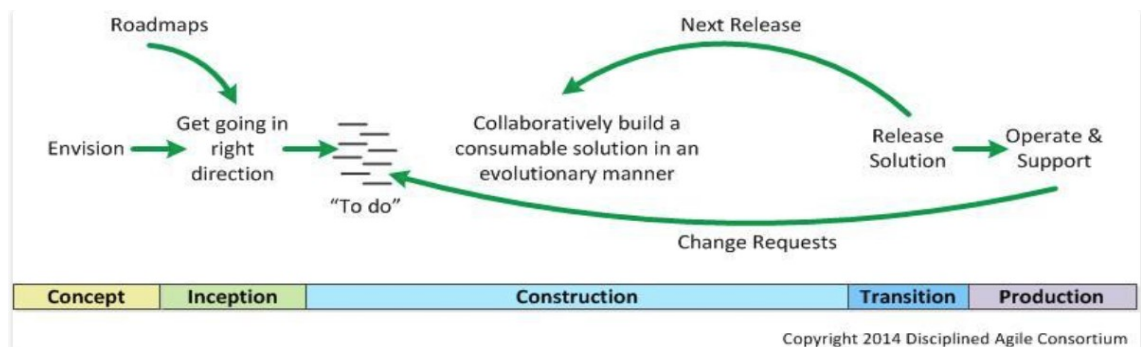
Kuva 19. Rakennusvaiheen tavoitteet

Siirtymisvaihe. DA tunnustaa, että kehittyneiden yritysten ketterissä hankkeissa, jotka tarjoavat toimivan ratkaisun sidosryhmille, ei ole useinkaan merkityksetön vaihe. Kehitystiimit sekä yritys kokonaisuutena tehostavat käyttöönottoprosessejaan, jotta tämä vaihe muuttuisi ajan myötä ja olisi osa jatkuvia käyttöönottostrategioita. (Ambler 2014)

Siirtymisvaiheen tavoitteita ovat.

- Varmistaa, että ratkaisu on valmis tuotantokäyttöön.
- Valmiin ratkaisun julkaisu.

Prosessi sisältää useita jatkuvan prosessin tavoitteita, jotka tapahtuvat koko toimituksen elinkaaren ajan. Näihin lukeutuvat kehitystiimin kasvattaminen tarvittaessa, tiimin tavoitteiden täyttäminen, rakennetun infrastruktuurin ylläpito ja kehittäminen, riskien tunnistaminen, tiimin prosessien ja työtapojen kehittäminen sekä tehtävien jatkuva koordinointi. (Ambler 2014)

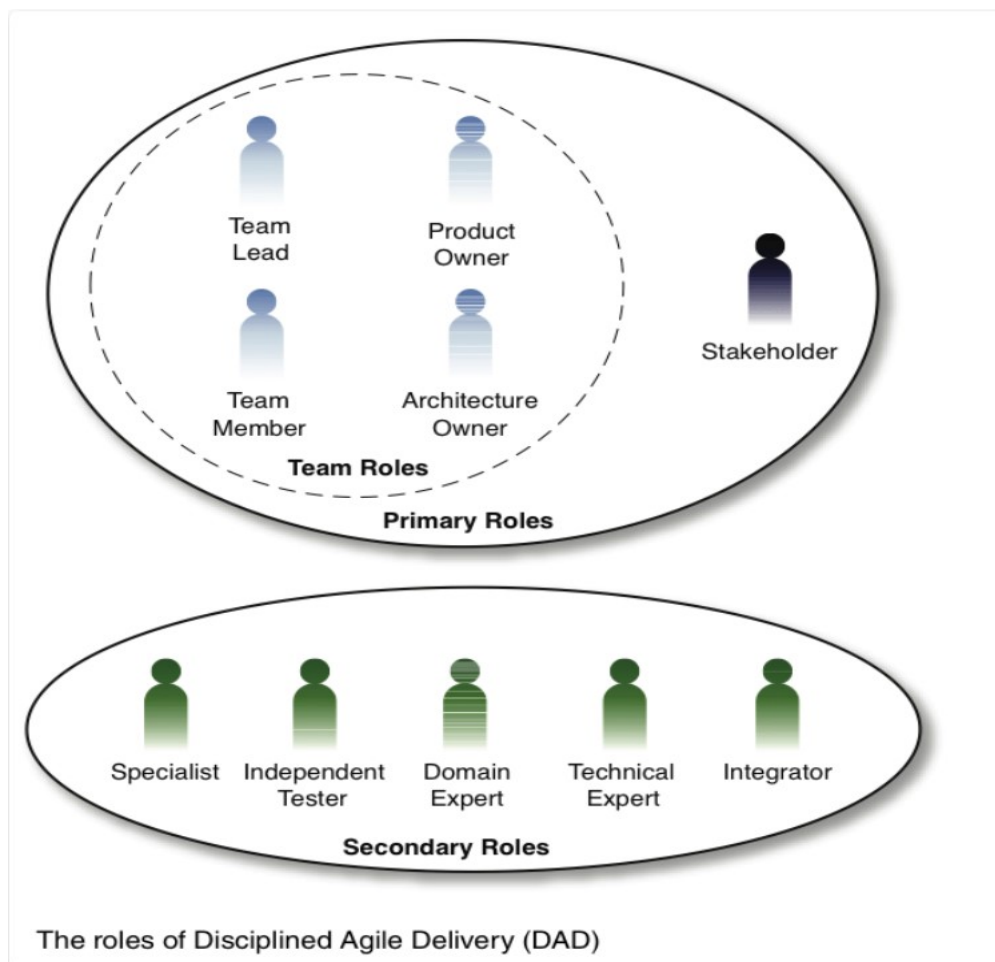


Kuva 20. DA elinkaaren laajennetut vaiheet

Kuva 20 osoittaa, että projektit sisältävät enemmän osavaiheita, kun edellä mainitut osavaiheet linjasivat. Ensinnäkin portfolionhallintaan liittyy ennalta analysoitavia hankekohtaisia näkökulmia, joissa mahdolliset hankkeet tai tuotteet tunnistetaan, vahvistetaan ja rahoitetaan alustusvaiheen käynnistämiseksi. Lisäksi organisaation ja teknisten vaiheiden etenemissuunnitelmat ohjaavat tiimin työtä. Siirtymisvaiheen jälkeen ratkaisu toimitetaan, käyttöön otetaan ja tuetaan tuotannossa. Lopulta, mahdollisesti vuosikymmenien käytön jälkeen, ratkaisu poistuu käytöstä. Jos tarkastelemme asioita IT:n näkökulmasta, yrityksen tasolla on myös ristikkäisiä tuotteita ja projekteja koskevia ongelmia, kuten yritysarkkitehtuuri, salkunhallinta, uudelleen käytön suunnittelu ja paljon muuta, joita DA-järjestelmä tukee laajennetulla mallilla. (Ambler 2014)

3.5.2 DAD vastuut ja roolit

Disciplined Agile Delivery (DAD) -kehiksessä ehdotetaan suuri joukko tehtäviä ketterille ratkaisuille. Näitä on olemassa kaksi ryhmää, jotka esitetään kuvassa 21.



Kuva 21. DAD roolit

Ensisijaiset roolit löytyvät yleisesti mittakaavasta riippumatta. Viisi ensisijaista tehtävää ovat.

- Sidosryhmät. Sidosryhmä on joku, jolle ratkaisun lopputulos vaikuttaa olennaisesti. Tältä osin sidosryhmä on enemmän kuin loppukäyttäjä. Sidosryhmä voi olla suora käyttäjä, epäsuora käyttäjä, käyttäjien johtaja, ylimmän johdon, operatiivisen henkilökunnan jäsen, projektin rahoittaja, henkilökunnan jäsen, tilintarkastajat tai kehittäjät, jotka työskentelevät muiden järjestelmien parissa, jotka integroivat tai ovat vuorovaikutuksessa ratkaisun kanssa. Tiimit tekevät tiivistä yhteistyötä sidosryhmiensä kanssa koko projektin ajan.
- Tiimin jäsen. Tiimin jäsenen rooli keskittyy varsinaisen ratkaisun tuottamiseen sidosryhmille. Tiimin jäsenet suorittavat testauksen, analysoinnin, arkkitehtuurisuunnittelun, ohjelmoinnin, kehityksen suunnittelun, arvioinnin ja monia muita toimintoja tarpeen mukaan koko projektissa. Voi olla, ettei jokaisella tiimin jäsenellä ole yhtäkään näistä taidoista, ainakaan vielä, mutta heillä on osa niistä ja he pyrkivät kehittämään osaamistaan ajan mittaan. Tiiminjäsenet määrittelevät tehtävät, arvioivat tehtävät, suunnittelevat tehtävät, toteuttavat tehtävät ja seuraavat tehtävien tilaa loppuun asti.
- Tiimin johto. Ketterillä projekteilla perinteisen projektipäällikön rooli muuttuu merkittävästi. Ketterä yhteisö on keskittynyt projekti- tai tiimijohtajuuteen tiiminhallinnassa. Tiimijohtaja on tiimin palvelija, luomalla ja ylläpitämällä edellytyksiä, jotka mahdollistavat ryhmän menestymisen. Tiiminvetäjä on myös ketterä valmentaja, joka auttaa pitämään tiimin keskittyneenä tekemään työtehtäviä ja täyttämään tavoitteensa ja sitoumuksensa, jotka he ovat antaneet tuoteomistajalle. Tiin johto toimii todellisena johtajana, joka helpottaa viestintää ja mahdollistaa tiimin optimoimaan prosessejaan, varmistaen, että tiimeillä on tarvittavat resurssit ja poistaa kaikki tiimin ongelmat ajoissa. Kun tiimit ovat itseohjautuvia, tehokas johtajuus on tärkeää menestyksen kannalta.
- Tuoteomistaja. Järjestelmässä, jossa on satoja tai tuhansia vaatimuksia, on usein vaikea saada liiketoiminnan edustajilta vastauksia vaatimuksiin liittyviin kysymyksiin. Tuoteomistaja on yksittäinen henkilö tiimissä, joka toimii asiakkaan ja tiimin linkkinä. Hän edustaa sidosryhmien tarpeita ja toiveita ketterälle tiimille. Selventää ratkaisun yksityiskohtia ja vastaa myös priorisoidusta työlistasta, jonka tiimi toteuttaa ratkaisuun. Tuoteomistajan toissijainen tavoite on edustaa ketterän tiimin työtä organisaation sidosryhmille. Tähän sisältyy ratkaisun esittely ja hankkeen tilan tiedottaminen keskeisille sidosryhmille.
- Arkkitehtuuriomistaja. Arkkitehtuuri on projektiriskin keskeisimpiä huolenaiheita. Tämän seurauksena DAD-kehityksessä nimenomaan sisältyy rooli arkkitehtuuriomistajasta. Arkkitehtuuriomistaja omistaa tiimin arkkitehtuuripäätökset ja helpottaa kokonaisratkaisujen luomista ja kehittämistä. Tiimin johto on usein myös arkkitehtuuriomistaja pienissä tiimeissä. Arkkitehtuuriomistaja on tyypillisesti tiimin vanhempi kehittäjä - ja joskus sitä voidaan kutsua tekniseksi arkkitehdiksi, ohjelmistosuunnittelijaksi tai ratkaisuarkkitehdiksi - on huomattava, että tämä ei ole hierarkkinen asema, jolle muut tiimin jäsenet raportoivat. Arkkitehtuuriomistajilla pitäisi olla tekninen tausta ja vankka käsitys liiketoimintalueesta.

Toissijaiset roolit näkyvät usein väliaikaisesti skaalausongelmien käsittelemiseksi. Toissijaisia rooleja on viisi:

- Asiantuntija. Vaikka useimmat ketterät tiimin jäsenet ovat erikoistuneita, joskus tarvitaan asiantuntijoita. Esimerkiksi suurissa tiimeissä yksi tai useampi ketterä liiketoiminta-analyttikko voi liittyä tiimiin, jotta voidaan vahvistaa liiketoiminnan asettamia vaatimuksia.
- Toimiala-asiantuntija. Tuoteomistaja edustaa monenlaisia sidosryhmiä, ei pelkästään loppukäyttäjiä, joten ei ole järkevää odottaa heidän olevan asiantuntijoita kaikissa toimialan haasteissa, mikä on erityisen tärkeää monimutkaisilla toimialoilla. Tuoteomistaja tuo joskus toimialanasiantuntijat työskentelemään tiimin kanssa, esimerkiksi verotusasiantuntija selittää vaatimuksen yksityiskohtia tai sponsoroiva johtaja tarkentaa projektin visiota.
- Tekninen asiantuntija. Joskus tiimi tarvitsee teknisiä asiantuntijoita tueksi. Näitä voivat esimerkiksi olla ketterä tietokannan pääkäyttäjä, joka auttaa tietokannan suunnittelussa ja testaamisessa. Käyttäjäkokemuksen (UX) asiantuntijan avulla suunnitellaan käytettävä käyttöliittymä tai turvallisuusasiantuntija antaa ohjeita turvallisen järjestelmän kirjoittamiseen. Teknisiä asiantuntijoita tuodaan tarpeen mukaan väliaikaisesti, jotta tiimi voi ratkaista vaikean ongelman ja siirtää taitoja yhdelle tai useammalle tiimin kehittäjälle. Tekniset asiantuntijat työskentelevät usein muissa tiimeissä, jotka ovat vastuussa yritystason teknisistä ongelmista tai ovat yksinkertaisesti asiantuntijoita, jotka konsultoivat kehitystiimiä.
- Itsenäinen testaaja. Vaikka suurimman osa testauksesta tekee tiimi itse, jotkut DAD-tiimit ovat tukeutuneet riippumattomaan testiryhmään, joka toimii rinnakkain ja validoi työtä koko elinkaaren ajan. Tätä riippumatonta testiryhmää tarvitaan tyypillisesti ketteryyden mittakaavassa monimutkaisten toimialojen kannalta, käyttämällä monimutkaista tekniikkaa tai ratkaisemaan lainsäädännön noudattamista koskevia kysymyksiä.
- Integraattori. Suuret DAD-tiimit, jotka on organisoitu useampaan pienempään tiimiin, ovat yleensä vastuussa yhdestä tai useammasta osajärjestelmästä tai ominaisuudesta. Mitä suurempi tiimi yleensä sitä suurempi ja monimutkaisempi järjestelmä on rakenteilla. Näissä tilanteissa tiimi voi pyytää yhdeltä tai useammalta henkilöltä integraattorin tehtävää, joka vastaa koko järjestelmän rakentamisesta koostuen eri osajärjestelmistä. Pienemmissä ryhmissä tai yksinkertaisissa tilanteissa arkkitehtuuriomistaja on tyypillisesti vastuussa integraation hallinnasta. Integraattorit toimivat usein läheisessä yhteistyössä riippumattoman testiryhmän kanssa suorittaen järjestelmällistä integraatiotestausta säännöllisesti koko hankkeen ajan. Tätä integraattorin roolia tarvitaan tyypillisesti vain monimutkaisten teknisten ratkaisujen osalta. (Ambler 2014)

Ensisijaiset roolit löytyvät tavallisesti DAD-ryhmistä riippumatta tiimin koosta ja toissijaiset roolit täytetään usein väliaikaisesti, jotta saadaan skaalausongelmat ratkaistua.

3.6 Scrum BI-projektin viitekehyksenä

Onnistuneen liiketoimintatiedon hallintaprojektin yksi tärkeimmistä tekijöistä on vaatimusmäärittely, joka on ketterien menetelmien suurin vahvuus. Perinteisten prosessimallien mukaisesti alkuvaiheessa määritellään vaatimukset ja projekti toteutetaan vaatimusten mukaisesti. Vaatimuksia tarkastellaan ketterissä menetelmissä jokaisen syklin alkuvaiheessa ja tässä vaiheessa pyritään myös ottamaan huomioon liiketoimintatarpeissa tapahtuneet muutokset. BI-projektin kaikkia vaatimuksia on vaikea tiedostaa ja määrit-

tää tarkasti ennen projektin aloitusta, joten tietovarasto on helpompi toteuttaa pienemmissä ja hallituissa osissa.

3.6.1 Ketterän BI-projektin edellytyksiä

Scrum Alliance on esittänyt listan edellytyksistä käyttää Scrumia BI-projekteissa. (Agile Alliance 2015)

Tärkein lähtökohta on valmius sprinttiin. BI-projektissa kytkeytyy seuraavia toimenpiteitä, että valmius sprintin suunnitteluun ja käynnistämiseen on mahdollista, kun tehdään uutta projektia.

Tuotteen ominaisuuslista

Tuotteen omistajalla tulee olla näkemys mitä ominaisuuksia tuotteeseen tulee. Valmistelut sprintin suunnittelukokoukseen on tehty. Jäädytetään ainakin kahden ensimmäisen sprintin ominaisuudet ja asetetaan ominaisuudet tärkeysjärjestykseen sekä julkaisuvalmiin tuotteen ominaisuudet on päätetty.

Ympäristön valmius

Yhteydet eri järjestelmien ja verkkojen sekä palomuurien välillä valmistellaan ennen aloitusta. Vähintään kehitysympäristö täytyy olla asennettuna ja valmiina kaikilta osin sekä kehitystyökalut asennettuina ennen toteutusvaihetta. Logistiikan ja kommunikaation vaatimukset kuten puhelin- ja videokokoukset määriteltä. Kaikki tiimin kehitystyötä tukevien ohjelmistojen tulee olla valmiina, kuten vaatimustenhallinta työkalut, ohjelmistojen versionhallinta sekä automatisoidut testiympäristöt.

Arkkitehtuurisuunnittelu

Korkeantason arkkitehtuurin tulee olla 80% valmis ennen aloitusta. Ohjelmoinnin käytännöt täytyy olla selvillä tiimille. Määriteltynä on myös ohjelmoinnin parhaat käytännöt ja standardit. Katselmointiprosessi on valmisteltu ja sovittu. Testauksen automatisoinnin valmiudesta tulee olla suunnitelma.

Arviointi

Käyttäjätarinat jaetaan kahteen kategoriaan, joita ovat tekniset ja liiketoiminnalliset tarpeet. Kaikki tekniset käyttäjätarinat hyväksytään projektin arkkitehdillä ja liiketoiminnan käyttäjätarinat tuoteomistajalla. Kehitystiimi ei suorita arviointiaan ennen kuin tekniset käyttäjätarinat on hyväksytty ja esitelty tiimille. Arkkitehtuurin käyttäjätarinat täytyy arvioida arkkitehdin toimesta luokitteluin helppo, keskivaikea, vaikea. Tiimi arvioi tarinapisteet perustuen arkkitehdin antamiin luokituksiin.

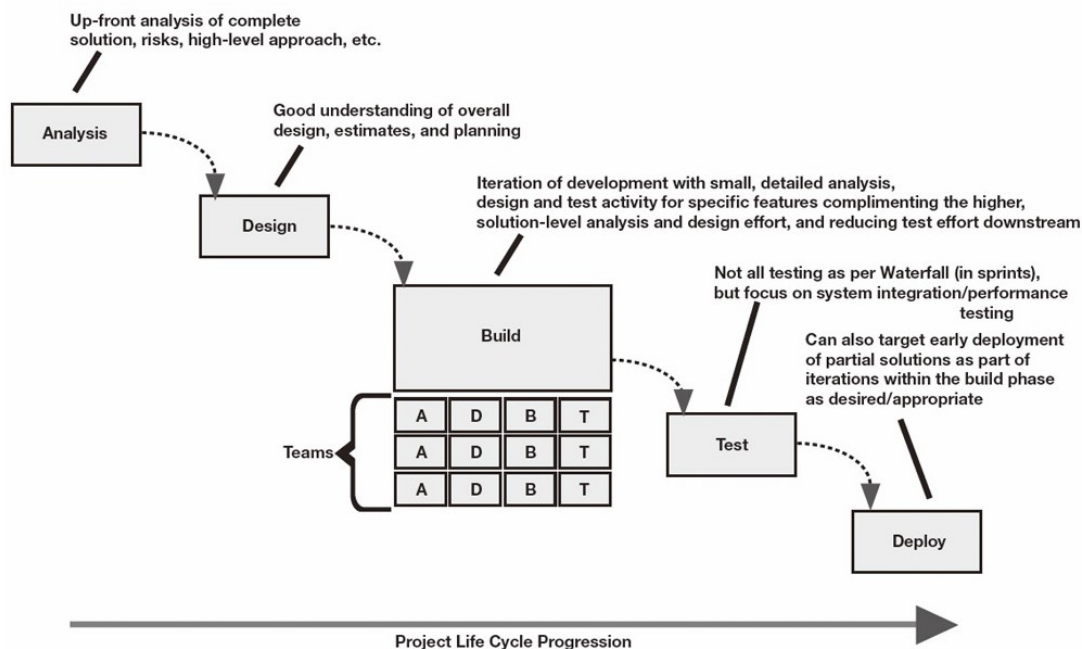
Tiimin rakenne ja valmius

Tiimin eri osa-alueiden asiantuntijat on kiinnitetty projektiin ja jokaisen kehittäjän sekä testaajan tarvitsemat kehitystyökalut on asennettu ja valmiina sprinttiin. Tiimiin tarvitaan tuotteen omistaja, Scrum Master, ETL-kehittäjä, raporttikehittäjä, projektin arkkitehti, tiedon arkkitehti, systeemisuunnittelija, systeemitestaaaja sekä tukitiimi.

Yhteenvedona voidaan todeta, että Scrum-viitekehys sopii BI-projekteihin, kun ottaa huomioon erilaiset roolit ja teknisen ympäristön valmiustilan ennen iteratiivista kehitysvaihetta. Kehitystä voi myös koordinoida useammassa kehityspotkussa, kun otetaan huomioon esimerkiksi kerrosten välinen valmiusaste. Tuotteenomistajan sitoutuminen on myös tärkeää. Sprintin työlistaa täytyy suojella ulkopuoliselta paineelta.

3.6.2 Water-Scrum-Fall

Water-Scrum-Fall esitellään Research Inc.:n artikkelissa *Water-Scrum-Fall Is The Reality Of Agile For Most Organizations Today* (West 2011). Artikkelissa mainitaan Water-Scrum-Fall -prosessin keskeinen ominaisuus - kehitystiimit toimivat vesiputousmallin keskellä soveltaen omaan kehitystyöhönsä Scrum-mallin mukaista toimintamallia. Kuvasssa 22 esitetään Water-Scrum-Fall-mallin vaiheistus. (Davis et al. 2014)



Kuva 22. Water-scrum-fall yhdistetty elinkaarimalli

Water-Scrum-Fall on hybridi lähestymistapa ratkaisun elinkaaren hallintaan, jossa yhdistyvät vesiputousmalli ja Scrum-kehitysmalli. Yleisesti ottaen kehitystiimi, joka käyttää vesiputousnäkökulmaa, pitää ohjelmistotuotteiden kehitysprosessia yhtenä suurena hankkeena. Projektin lopussa julkaistaan ratkaisu loppukäyttäjille. Tyypillisesti yrityksen tuoteomistaja näkee vain valmiin tuotteen. Sitä vastoin kehitystiimi, joka käyttää Scrumia tai jotain muuta ketterää menetelmää, lähestyy samaa kehityshanketta pienen projektin näkökulmasta, joita kutsutaan iteraatioiksi. Toimiva ohjelmisto julkaistaan säännöllisesti inkrementeissä, kunnes koko lopputuote on valmis. Tyypillisesti projektin omistajalla on aktiivinen rooli koko prosessissa ja omistaja liittyy myös kehitystiimin retrospektiiviin jokaisen julkaisun jälkeen.

Joustava lähestymistapa, joka kattaa sekä perinteiset että ketterät kehittämisperiaatteet, mahdollistaa kehitystiimien käyttää ketteriä käytäntöjä ja tekniikoita, jotka vastaavat parhaiten ongelman nykytilanteen ratkaisemiseen. Monet organisaatiot käyttävät ketteriä periaatteita ja viestintätekniikoita päivittäisessä tuotekehityksessään, kun taas perinteisiä vesiputousmenetelmiä käytetään projektin edistymisen suunnittelussa, seurannassa, budjetoinnissa tai dokumentoinnissa.

Monet tiimit ovat omaksuneet Scrumin-perusperiaatteet, kuten päivittäiset kokoukset, tuoteomistajan ja Scrum Masterin roolit sekä Scrumin suunnittelu- ja retrospektiivikäytännöt. Scrumin menestys voi liittyä moniin asioihin, mutta erityisesti tiimiin ja tiimien dynamiikkaan kohdistuva vahva keskittyminen on herättänyt monia ihmisiä, jotka kokevat, että perinteiset lähestymistavat eivät huomioi ihmisiä prosessin takana.

West (2011) kirjoittaa, että monilla organisaatiolla on pitkä historia perinteisten hallintatyökalujen käytöstä. Ratkaisun kehitysoikeutta varten laaditaan budjetteja, suunnitelmia, arkkitehtuurisuunnitelmia ja vaatimusmäärittelyjä sekä riskilistoja, jotka rajaavat ketterien menetelmien kannalta tärkeää liikkumavaraa. Vaatimukset näiden työkalujen ja menetelmien käyttöön tulevat usein projektiorganisaation ulkopuolelta, jolloin niihin on vaikeampi vaikuttaa.

Westin (2011) tekemien havaintojen mukaan Water-Scrum-Fall-malliin ajautuminen voidaan nähdä myös seurauksena siitä, että ketteryys nähdään alun perinkin kehitystiimin, ei koko organisaation menetelmänä ja ominaisuutena. Ketterien menetelmien käyttöönotto tapahtuu toteuttajakeskeisesti. Kun muutos tapahtuu toteutusjaksolla, kohdistuu se samalla luonnollisesti niihin osiin prosessista, jotka kehitystiimit kokevat omikseen. Toisaalta toteuttajilta puuttuu yleensä aito mahdollisuus vaikuttaa organisaation toimintaan oman toimialueensa ulkopuolelta. Vaikka tahtotila ketterien menetelmien käyttöönottoon löytyisi, ei muutoksen edistämiseen läpi organisaatioon ole käytännössä mahdollisuutta, ellei muutoksen eteenpäin viejä ja ajava voima omaa organisaatiossa sellaista asemaa, jossa on mahdollista vaikuttaa toimintamalleihin kaikilla organisaation tasoilla. Tämä kuvaa hyvin asiakkaan sitoutumisen tarpeen, jota ketterät menetelmät tarvitsevat sekä halun aktiiviseen osallistumiseen. Organisaatio mielellään

ottaa mallin hyvät puolet käyttöön mutta onko ketterillä menetelmillä organisaation johdon tuki takana.

3.7 Mitä etuja ketteristä menetelmistä on BI-hankkeissa

Asiakastyytyväisyys on suuri arvo, jota tavoitellaan tällä hetkellä tiukasti kilpailluilla BI-markkinoilla. Yksi tekijä, jolla asiakas pidetään tyytyväisenä, on nopea toimitusaika toimivalla sovelluksella, joka tuottaa liiketoiminnalle lisäarvoa uudella toiminnallisuudellaan. Ketterissä menetelmissä myös hyväksytään tosiasia, että muutoksia saattaa tulla myös projektin loppupuolella. Muutokset otetaan mukaan mahdollisimman nopeasti seuraavaan iteraatioon, mikäli tämä muutos on asiakkaan kannalta tärkeä. Ketterissä menetelmissä pyritään toimittamaan julkaisu mahdollisimman nopeissa sykleissä – noin kahden viikon tai kuukauden välein. Tavoitteena on jokaisen ominaisuuden jälkeen, että saatavilla on toimiva sovellus. Tasainen työ- sekä toimitustahti on myös kehittäjien, asiakkaan sponsoreiden ja loppukäyttäjien mieleen. Loppukäyttäjät voivat opetella ja heille voidaan kouluttaa uudet ominaisuudet pienemmissä osissa. Ketterien menetelmien etuna pidetään tiivistä kommunikaatiota liiketoiminnan ja kehittäjien kanssa. Läpinäkyvyys paranee, kun toimittaja ja tilaaja keskusteleivat kasvotusten sekä pitävät säännölliset tilannekatsaukset. Työn edistyminen ja seuraavat työvaiheet päivitetään päivittäin ja työlistaa hallitaan säännöllisesti.

4. TUTKIMUS

Monissa yrityksissä on projekti tai projekteja käynnissä. Ne voivat koskettaa yhtä osastoa tai toimia useamman osaston tai liiketoimintayksikön yli ja näin ollen voidaan ajatella, että projektit ovat osa liiketoimintaa tai vähimmilläänkin vaikuttavat liiketoiminnan tarpeisiin ja tulevaan suuntaan. Tyypillisesti liiketoiminta sisältää kaksi laajaa osaa. Siihen kuuluu operatiivinen osa, joka hoitaa päivittäiset aktiviteetit esim. myynti, tuotanto, jakelu, laskutus, logistiikka tai toimitus. Toinen osa on projekti, joka keskittyy liiketoiminnan tulevien haasteiden ratkaisemiseen. Se saattaa sisältää tuotekehitystä, markkinointiohjelmat sekä liiketoimintaprosessien kehittäminen. Yleisesti ottaen tuotteet, palvelut, prosessit luodaan ja kehitetään projekteissa ja sitten siirretään operatiivisen toiminnan käyttöön.

Ketterässä strategiassa nähdään projektit samasta näkökulmasta liiketoiminnan kanssa. Projektit ja liiketoiminnan päätöksentekoprosessit integroidaan, jolloin saavutetaan parhaalla mahdollisella tavalla liiketoiminnalle asetetut tavoitteet.

4.1 Kohdeyritys

Kohdeyritys, johon tutkimus on toteutettu, toimii liiketoimintatiedon hallinnan liiketoimintaympäristössä. Kohdeyritys on teknologiariippumaton konsulttitalo, jonka työntekijämäärä on noin 30 henkeä. Yrityksen asiakaskuntaan kuuluu toimijoita eri toimialoilta pienistä suuriin yrityksiin. Projektien sisältö ja koko vaihtelevat suuresti vaikkakin palvelut kohdistuvat suurelta osin liiketoimintatiedonhallintaan. Pienimmillään kohdeyrityksen konsultti saattaa toimia kohdeyrityksen projekteissa asiakkaan haluamassa roolissa. Suurimmillaan toimeksiannot saattavat kestää useita kuukausia ja projektiryhmä koostuu useista kohdeyrityksen asiantuntijoista. Toiminta on jaettu konsultointipalveluihin sekä kehitys- ja tukipalveluihin.

Kohdeyrityksen organisaatorakenne on matala ja se koostuu erikokoisista projektiryhmistä. Projektiryhmän asiantuntijat kootaan asiakkaan toimeksiannon mukaan, jotta saadaan tarvittava osaaminen projektinjohdosta tarvittaviin teknisiin kyvykkyyksiin katettua. Kohdeyrityksen konsultointipalvelut voidaan jakaa seuraaviin osa-alueisiin.

- Konseptit ja tiekartat
- Teknologiavalinnat
- Ratkaisujen terveystarkastukset
- Analytiikka ja ennustaminen
- Tiedolla johtaminen

- Datapohjainen liiketoiminta

Kehitys- ja tukipalvelut on jaettu seuraaviin toimintoihin.

- Järjestelmäkonseptit ja migraatiot
- Business Intelligence-projektit
- CPM-projektit
- DW, DWA ja Big Data
- Raportointi
- Jatkuvat palvelut

Kohdeyrityksen palveluita ja ryhmittelyä muutetaan liiketoimintatarpeiden muuttuessa sekä strategiamuutosten yhteydessä.

4.2 Tutkimusongelma

Tutkimuksen toteuttamisen aikaan kohdeorganisaatio toimitti useita erikokoisia BI-projekteja useille erikokoisille asiakkaille. Asiakkaat toimivat laidasta laitaan eri toimialoilla ja asiakkaina oli niin yksityisiä kuin julkisia toimijoita. Usealla suurella yrityksellä BI oli kehitetty ja integroitu osaksi tukemaan yrityksen liiketoimintaa ja heillä oli käytössään vakiintuneita sekä kehittyviä projektinhallintamenetelmiä niin perinteisistä kuin ketteristä menetelmistä. Pienillä tai keskisuurilla yrityksillä ei välttämättä ollut käytössä vakiintuneita projektinhallinta malleja, joten kohdeyrityksessä nähtiin tarve kehittää ketterä malli tukemaan projekteja sekä osaksi myyntityötä.

Tutkimusongelmana oli: miten saadaan luotua kattava projektinhallintamenetelmä, joka kuitenkin sallii joustavuuden projektin laajuudesta, sisällöstä tai toimialasta riippuen? Käytetty tutkimusmenetelmä oli suunnittelutieteellinen, ja tutkimuksen konstruktion tuloksena syntyi projektinhallintamenetelmä BI-projekteihin, voitiin tutkimusongelman tarkentaa edelleen kysymykseen: millainen projektinhallintamalli täyttää kohdeorganisaation vaatimukset projektin arvoa tuottavien ratkaisujen sekä laadullisten vaatimusten varmistamiseksi? Tutkimusongelman asettaminen oli kohtalaisen ongelmatonta, koska tarve projektinhallintamenetelmälle oli tunnistettu useiden asiakasprojektien yhteydessä, kun menetelmää ei ollut asiakkaalta ennalta määritetty.

5. TYÖN TULOKSET JA HAVAINNOT

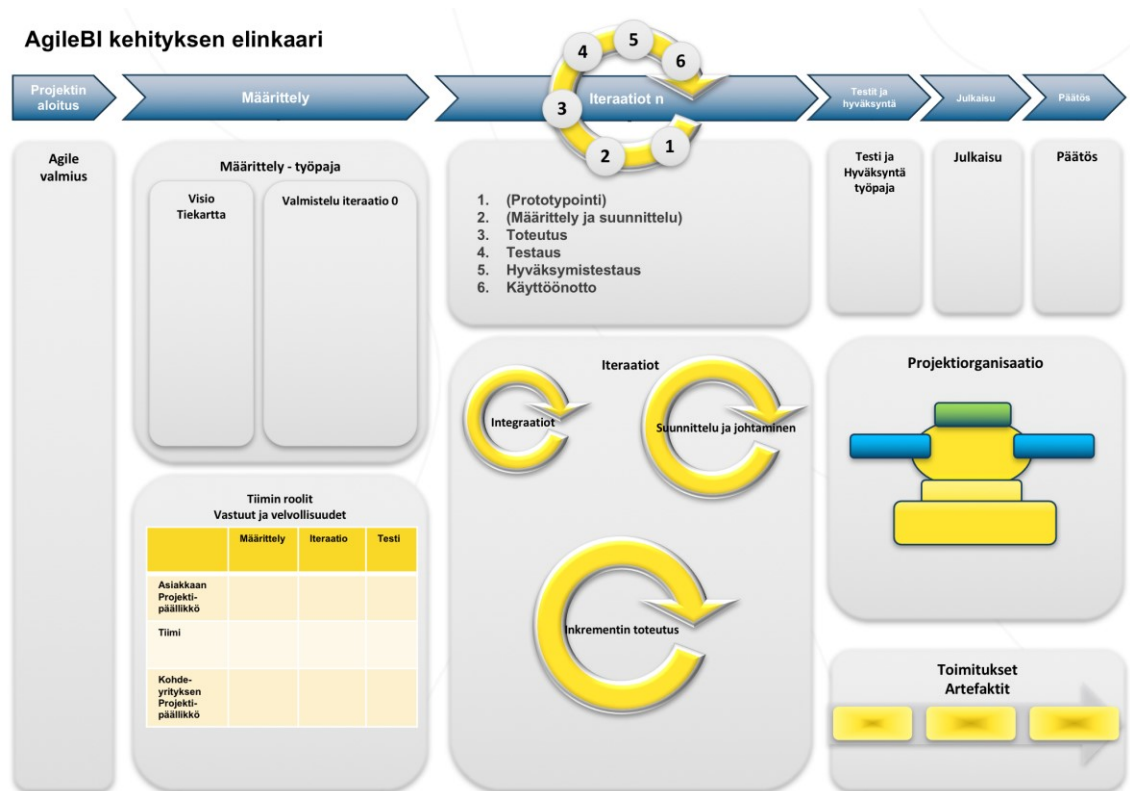
Tutkimuksessa kehitettiin tuotoksena projektinhallintamalli, jolla kyetään hallitsemaan ja johtamaan erilaisissa asiakastoimeksiannoissa syntyneitä projektinhallinnan haasteita. Tässä luvussa käydään kohdeyritykselle kehitetty projektinhallintamalli vaiheittain läpi alkaen projektin aloitusvaiheesta projektin sulkemiseen. Rakenteen pohjana on vertailu vaiheistetun toimintamallin ja ketterän toimintamallin eroista sekä pyritty hahmottamaan hyödyt kohdeyritykselle tuotettuun malliin. Näkökulmia on esitelty taulukossa 2.

Taulukko 2. Toimintatapojen näkökulmia ja eroja

	Vaiheistettumalli	Ketterä malli	Hyödyt
Projektin aloitus	Rinnakkaiset hankkeet syövät toisiltaan resurssit	Toteutuksessa on vain se määrä vaatimuksia, joka pystytään viemään läpi	Fokus ja tavoitteellisuus kasvavat Resurssien optimaalinen käyttö
Koko ja läpimenoaika	Hankkeiden aloittaminen ja arvonsaaminen kestävä, hankkeet ovat suuria	Kehityskohteiden pilkkominen pienempiin ja helpommin hallittaviin osiin analyysin ja palvelusuunnittelun avulla Kun investoinnit tehdään pienemmissä osissa, helpottuu jatkuvan ketterän kehittämisen lisäksi myös lopputulosten ostaminen Lopputulosten hyväksyminen jatkuvasti, ei ainoastaan lopuksi	Nopeampi läpimenoaika Aikainen arvontuotto Nopeampi reagointi muutoksiin
Vaatimusten ja tavoitteiden kuvaaminen	Asetetut tavoitteet ohjaavat ratkaisun toteutusta rajallisesti	Otetaan käyttöön työtavat, jotka auttavat pilkkomaan tavoitteet ja vaatimukset siten, että se ohjaa toiminnallisuuksien ja ratkaisun toteutusta Visiot, julkaistavat kokonaisuudet, asiakashyötyä tuovat toiminnallisuudet	Valistunut päätöksenteko Yhteiset päätökset fokusoivat tekemistä
Päätöksenteko	Investointipäätöksessä kiinnitetään laajuus ja lopputulokset	Investointipäätöksen yhteydessä kiinnitetään minimitalvoitteet Sisältö joustaa, kustannukset kiinnitetään	Päätöksen epävarmuus pienenee Läpinäkyvyys kasvaa

Arkkitehtuuri	Arkkitehtuurityö painottuu ratkaisun valmisteluun ja investointiin	Arkkitehtuuri otetaan huomioon riittävän varhaisessa kehittämisen vaiheessa jo portfoliotasolla Arkkitehtuurityö tulee osaksi kehitystyötä tarkentuen jatkuvasti	Hallittu arkkitehtuurin koko elinkaaren kustannukset Arkkitehtoniset kyvykkyudet tukevat arvontuottoa
Kustannukset	Hankkeen budjetointi ja toteuman seuranta yhdessä	Adaptiivinen kustannussuunnittelu ja –seuranta Erotetaan budjetointi ja toteuman seuranta Arvioidaan kunkin investoinnin vaikutus ja laajuus, tähän asti tehdyt investoinnit, tähän asti saavutetut tulokset ja tulevaisuuden kehitys	Investoidaan sinne, mistä saadaan parhaiten hyötyä Mahdollistaa sisällöllä johtamisen
Johtaminen	Johto ohjaa sisältöä yksityiskohtaisesti	Johto ohjaa prioriteeteilla ja investointiteemoilla tuotekehitystä aktiivisesti Prioriteettimuutokset ja uudet linjaukset muuttavat nopeasti kehityksen suuntaa ja painopistealueita; nopea prioriteettien ohjaama päätöksenteko Läpinäkyvyys kaikkeen tekemiseen; nopea palaute ja ongelmien ratkaisu	Nopea kyky reagoida muutoksiin
Tiekartta	Tiekartta tehdään vain hankkeen alussa tai työpuutteellista	Tiekarttaa päivitetään jatkuvasti ja tarkennetaan ajan kuluessa hyödyntäen viimeisintä tietoa Suunnitellaan asioita eteenpäin, lähiaika näkyy paremmin ja tulevaisuus tarkentuu ajan kuluessa	Selkeä kuva, miten arvontuotto realisoituu
Arvovirrat	Teknisten ratkaisukohtaisten vaatimusten määrittely	Tunnistetaan arvovirroilta haluttavat liiketoimintahyödyt Arvovirroissa kehitettäviä uusia asioita ja vanhojen jatkokehitystä tarkastellaan kokonaisuutena	Tekeminen fokusoituu Kyky priorisoida arvon mukaan

Ketterän mallin hyötyjä pohdin seuraavien tekijöiden kautta. Tekijät olivat projektin aloitus, koko ja läpimenoaika, vaatimusten ja tavoitteiden kuvaaminen, päätöksenteko, arkkitehtuuri, kustannukset, johtaminen, tiekartta eli pidemmän aikavälin tavoitteet sekä organisaation arvovirrat. Hyötyjä löytyi paljon ketteryyden puolesta mutta on tunnustettava, että näkökulmia voidaan tutkimukseen valita subjektiivisesti ja mallin kehittäjän näkemykset ja kokemuspohja vaikuttavat hyötyjen tulkitsemiseen.



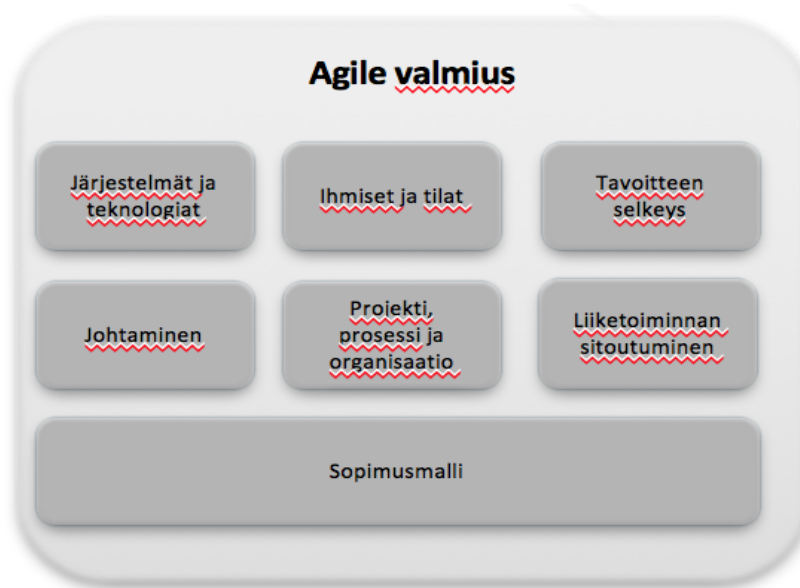
Kuva 23. Ketterän kehityksen elinkaari

Kuva 23 esittää mallinnetun ketterän kehityksen elinkaaren yleisellä tasolla. Projekti käynnistetään tarjouksen hyväksynnän tai asiakkaan tekemän tilauksen jälkeen, kun kohdeyritys on valittu ratkaisun toimittajaksi. Sen jälkeen käynnistyy projekti, jossa synnytetään asiakastarpeet kattava ratkaisu. Ketterän kehittämisen elinkaareen on yllätasolla valikoitunut seuraavat vaiheet: projektin aloitus, määrittely, tiimin roolit, vastuut ja velvollisuudet, iteraatiot, testit ja hyväksyntä, julkaisu ja projektin päätös.

5.1 Projektin aloitus

Projektin ensimmäistä vaihetta mallissa kuvataan projektin aloitukseksi. Projektin aloitusvaiheessa varmistetaan, että onko toimeksiantajalla omaa projektinhallintamallia. Jos on, niin käytetään pääsääntöisesti sitä. Tarjouksessa esitetään myös kohdeyrityksen ketterän kehityksen malli. Projektimallin käytön onnistumisen edellytyksiksi on tunnistettu seuraavat tekijät, joita kutsutaan termillä ”Agile-valmius”. Alustavasti arvioidaan projektin monimutkaisuus, koko ja kesto. Seuraavaksi keskustellaan ja arvioidaan, onko keskeisillä sidosryhmällä (etupäässä asiakkaalla) ennakkoon selkeät vaatimukset ja kirjas käsitys lopputuloksesta ennen hankkeen käynnistämistä. Sidosryhmän kanssa käydään läpi, että muuttuvatko vaatimukset sekä laajuus mahdollisesti projektin aikana ja odotetaanko edellä mainittujen asioiden muuttuvan projektin elinkaaren aikana. Valinnan kannalta on ensiarvoisen tärkeää, että asiakas sitoutuu projektiin niin sisällön, muutosten, testien kuin käyttöönoton osalta. Toimittaja joutuu arvioimaan, että ovatko projektiin osallistuvat henkilöt ja tiimi mukautuva, taitava, koulutettu alati vaihtuviin vaa-

timuksiin, dynaamiseen ja nopeaan toimitusmalliin. Projektin käytössä olevat teknologiat ja työkalut tulee myös arvioida henkilöstön osaamisen kannalta. Asiakkaan käytössä olevan ympäristön suhteen valintaan vaikuttaa liitännöiden laajuus ja integrointi ulkoiisiin prosesseihin sekä järjestelmiin. Monimutkaiset ympäristöt vaativat useamman rinnakkaisen kehityspolun, jotta töiden edistymistä ei joudu odottamaan työvaiheiden kesken. Ratkaisun haluttu julkaisu ja käyttöönottoaikataulu vaikuttavat myöskin päätöksen. Liiketoiminnan asettama ulkoinen paine ja tavoitetilä ohjaavat päätöstä siitä, että julkaistaanko ratkaisu esimerkiksi kuukauden vai puolen vuoden päästä. Projektin alustavissa keskusteluissa selvitetään myös projektilta vaadittava dokumentoinnin taso.



Kuva 24. Agile-valmius

Asiakkaan valmiudessa ketteriin menetelmiin selvitetään kuvassa 24 mainitut tekijät. Projektiin valittavat järjestelmät ja teknologiat sekä työkalut valitaan ensisijaisesti tukemaan parasta mahdollista ratkaisua mutta niiden tulee taipua ketterän kehittämisen tavoitteisiin. Ketterässä ajatusmallissa kehitystiimillä on valtaa valita kehityksen ja kommunikaation vaatimat työkalut. Organisaation johtamistapa analysoidaan ja selvitetään. Perinteisessä mallissa varaudutaan eri ongelmiin ja tehdään laaja analyysi koetuista riskeistä, kun taas ketterässä johtamisessa reagoidaan muutoksiin mahdollisimman nopeasti ja joustavasti. Ketterässä johtamisessa kommunikaatio ja vuorovaikutus ovat tärkeässä roolissa ja sidosryhmien kanssa halutaan kommunikoida jopa päivittäin. Johtaminen ja vastuullisuus ovat yhteinen prosessi johon koko kehitystiimi osallistuu projektipäällikön kanssa, kun taas perinteisessä mallissa johtaminen kuuluu projektipäällikön vastuusiin. Jaettu johtajuus johtaa itseohjautuviin tiimeihin, joilla on vastuu tiimin työn suhteen ja näin ollen tiimin jäsenet sitoutuvat yhteisiin tavoitteisiin ja ottavat vastuun toiminnastaan. Asiakkaan ratkaisun tavoitteet selvitetään työpajassa. Työpajan keskusteluiden ja ryhmätöiden jälkeen nähdään, kuinka selkeä tavoite asiakkaalla on ratkaisun suhteen. Ratkaisu toteutetaan inkrementaalisesti, mikäli tavoite on epäselvä ja

tavoitetta halutaan selventää kehityksen myötä. Työpajassa päätetään myös, miten paljon asiakkaan liiketoiminnan edustajilla on aikaa käytettävissä ratkaisun katselmointiin ja vaatimusten selkeyttämiseen tarvittaessa.

5.2 Määrittely

Vaiheittain etenevässä projektimallissa valtaosa vaatimuksista kootaan heti hankkeen alkuvaiheen vaatimusmäärittelyvaiheessa. Tarkoituksena on määritellä, mitä on toimitettava niin, että kaikki tietävät laajuuden ja yksityiskohtaisen kuvauksen siitä, mitä projektissa toteutetaan. Toimittajan asiantuntija viettää tyypillisesti sovitun ajanjakson järjestelmän loppukäyttäjien tai heidän edustajiensa kanssa kooten heidän liiketoiminta vastuualueen vaatimukset ja miten ne täytetään kehitettävässä ratkaisussa. Tämä kaikki johtaa dokumenttien ja kaavioiden tuottamiseen sekä toiminnalliseen määrittelyyn ja käyttötapausten kirjaamiseen. Nämä ovat määritelmä siitä, mitä projektissa rakennetaan ja jota seurataan koko hankkeen ajan. Menestyneimmissä vaiheittain edistyvissä projekteissa määritelmät pidetään ajan tasalla koko projektin ajan, kun muutoksia tapahtuu.

Ensisijainen kysymys liittyy vesiputouksen aikatauluun. Vesiputousprojekti voi kestää useita kuukausia tai vuosia ennen sen toimittamista. Kun toimittaja asettaa itsensä loppukäyttäjän asemaan ja pohtii loppukäyttäjän asemasta, että miltä liiketoimintaympäristö näyttää vuoden kuluttua. Tästä voidaan johtaa oletus missä nykyhetkessä tunnistamatta jäänyt vaatimus joutuu odottamaan toista vuotta, kun alkaa seuraava projekti, joka toteuttaa seuraavan vaiheen vaatimukset. Useasti on asiakkaan vaikea tunnistaa nykyhetkessä mitkä liiketoimintaprosessit tulevat olemaan tärkeimmät kuluvan vuoden aikana. Monesti loppukäyttäjä tarvitsee liiketoimintahallinnan ratkaisun käyttöönsä, jotta voi tunnistaa tärkeimmät prosessit. Asiakas joutuu tunnistamaan ja ennakoimaan ympäristön muutokset, jotka saattavat vaikuttaa liiketoimintaan seuraavan vuoden aikana. Liiketoimintaympäristön asettamien tekijöiden vuoksi saattaa olla järkevää harkita ketteriä iteratiivisia projektinhallintamalleja. Edellä mainittuja tekijöitä ovat muun muassa markkinoilla vallitseva kilpailu, lainsäädännölliset muutokset, yhteiskunnan kriisit, jotka lisäävät liiketoimintaympäristön epävakautta. Usein tulee vastaan myös toimittajan ja asiakkaan välisiä taisteluja, joissa vaaditaan, että jonkin vaatimuksen on oltava ja mikä olisi pitänyt olla jne. pitkissä projekteissa. Tämä jättää arvet molemmin puolin ja usein tarkoittaa onnettomia asiakkaita.

Vesiputousmallissa vietetään projektin ensimmäiset kuukaudet analysoimalla vaatimuksia ja kirjoittamalla vaatimusmäärittelyä. Tuolloin asiakas on käyttänyt kohtuullisen määrän budjettiaan, ja ainoa asia mitä saadaan valmiiksi, on määrittelydokumentti ja ehkä jotkut varhaisen vaiheen kaaviot. Ovatko nämä arvokkaat artefaktit? Todellisuudessa ne eivät lisää liiketoiminnan arvoa. Asiakas ei voi ansaita investointinsa tuottoa näillä dokumenteilla, vaan tarvitsee liiketoiminnan hallintaan ratkaisuja arvon tuottamiseen. Dokumentit eivät tarjoa yhtä toiminnallisten ratkaisujen linjaa, vaan ne toimivat vain välttämättömänä askeleena matkalla siihen toimivaan ratkaisuun.

Ketterässä iteratiivisessa hankkeessa asiakkaan ei tarvitse käsitellä kerralla kaikkia heidän tarpeitaan yksityiskohtaisesti tulevaisuuteen. Sen sijaan hankkeen laaja kuva määritellään etukäteen ja sitten jokaisella iteraatiolla analysoidaan tarpeita, jotka tukevat kyseisen ratkaisun toimittamista ominaisuus kerrallaan tärkeysjärjestyksessä muuttuvassa liiketoimintaympäristössä. Tärkeintä on, että asiakas saa jokaisen iteraation tulokset erittäin nopeasti, mikä tarkoittaa, että heillä on oikeus kertoa, mikä on heille tärkeintä nykyhetkessä, tietäen, että hankkeen edetessä he voivat tarkentaa vaatimuksiaan. Jos osoittautuu, että vaatimusta ei koskaan tarvita, he säästävät resursseja tekemättä yksityiskohtaista analyysia, puhumattakaan tarvittavasta työstään, testauksesta ja ylläpidosta. Lisäksi, jos uusi liiketoimintavaatimus ilmaantuu, asiakas voi uudelleen priorisoida toiminnot ilman kalliita muutospyyntöjä. Esimerkiksi jos kilpailija tulee ulos uudella ominaisuudella, asiakas kykenee vastaamaan kilpailuun nopeasti, ja asettaa muut vaatimukset pienempään arvoon tärkeysjärjestyksessä. Lopuksi asiakas ja kehitystiimi oppivat yhä enemmän projektin edetessä, varsinkin kun ratkaisu on oikeassa käytössä. Tämä tarkoittaa, että heillä on enemmän kokemusta ja he voivat tuottaa parempia vaatimuksia. Asiakas on esimerkiksi oppinut lisää vaatimuksistaan, kun hän on nähnyt käytössä olevan ratkaisun, ja heillä on parempi käsitys siitä, mikä toimii parhaiten vaatimusten pohjalta vuorovaikutuksessa kehittäjien kanssa.

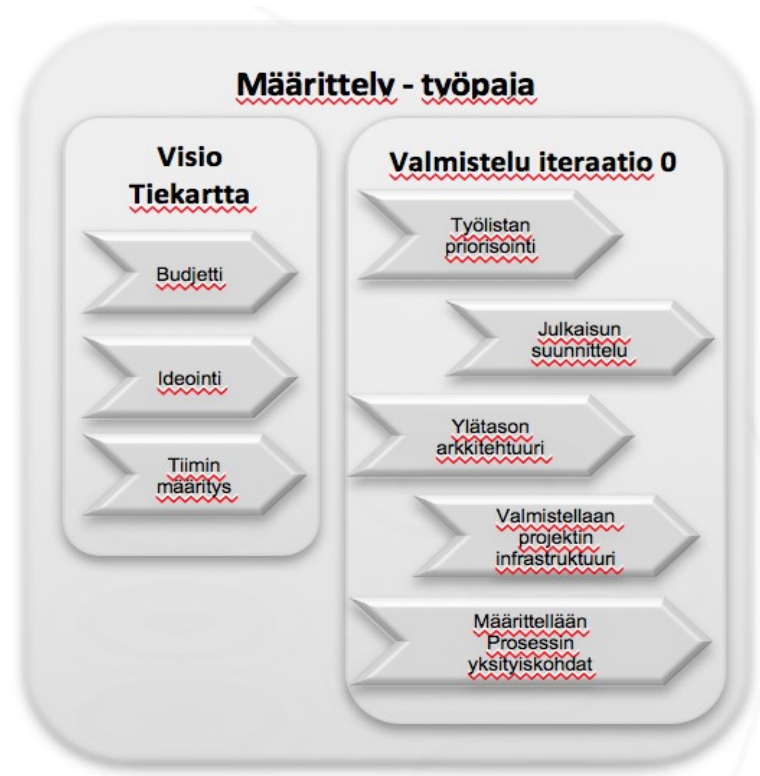
MÄÄRITTELYVAIHEEN TYÖPAJA

Ennen hankkeen aloittamista luodaan korkean tason visio siitä, mitä hankkeelta halutaan pitkässä juoksussa. Visio vastaa kysymyksiin *”miksi ratkaisu tehdään”* ja *”mitä hankkeen lopputuloksena saadaan kasvattamaan liiketoiminnan arvontuottoa”*.

Määrittelyvaiheen tavoitteita ja tehtäviä on esitetty kuvassa 25. Tiekartta on kokonaiskuva tuotteen vaatimuksista ja arvokas väline ratkaisun kehityksessä, suunnittelussa ja organisoinnissa. Tuotteenomistaja luo tiekartan kehitystiimin avulla. Etenemissuunnitelmaa käytetään luokittelemaan vaatimukset, priorisoimaan ne ja määrittelemään aika-tila ominaisuuksien toteuttamiseen. Kun luodaan ensin tiekartta, luultavasti aloitetaan suurista, korkeantason vaatimuksista. Tiekartassa vaatimukset ovat todennäköisesti kahdella eri tasolla teemoittain sekä ominaisuuksittain. Teemat ovat loogisia korkeantason luokitteluja ominaisuuksista ja vaatimuksista.

Ominaisuudet ovat ratkaisun osia erittäin korkealla tasolla. Ominaisuudet kuvaavat asiakkaalle uuden ominaisuuden ratkaisussa, kun toiminto on valmis. Kun luodaan tiekarttaa, tunnistetut ominaisuudet alkavat muodostaa tuotteen ominaisuuslistan - täydellinen luettelo ratkaisun sisällöstä menemättä yksityiskohtaiselle tasolle.

Kun tunnistetaan ratkaisulle asetettuja vaatimuksia, työskennellään kehitystiimin kanssa ryhmittelemällä vaatimukset teemoihin. Työpaja toimii hyvin vaatimusten ryhmittelyssä, vaatimusten ideoinnissa ja tunnistamisessa. Ominaisuuksia ryhmitellään esimerkiksi käyttötarkoituksen mukaan, teknisellä samankaltaisuudella tai liiketoiminnanalueen tarpeella.



Kuva 25. Määrittelyvaihe

Kysymykset, jotka kannattaa ottaa huomioon, kun ryhmitellään vaatimuksia.

- Mihin tarpeeseen asiakkaat käyttäisivät ratkaisua?
- Onko vaatimuksella lisäominaisuuksia, jotta saadaan toimiva kokonaisuus?
- Tunnistaako kehitystiimi teknisiä suhteita tai riippuvuuksia?

Aluksi ratkaisun vaatimukset tunnistetaan ja järjestetään loogisiksi ryhmiksi. Sen jälkeen arvioidaan ja tilataan vaatimukset. Muutamia asioita, jotka on tunnettava tässä vaiheessa, ovat vaatimuksen toteuttamisen helppous tai vaikeus sekä arvio vaatimuksen tarvitsemasta työmäärästä. Priorisointi edellyttää, että vaatimuksen tärkeys määritetään suhteessa muihin vaatimuksiin. Arvotuotto tarkoittaa sitä, kuinka hyödyllinen tietyn ratkaisun vaatimus voi olla kyseisen ratkaisun luomaan arvoon organisaatiossa. Hyvä on tunnistaa myös vaatimusten väliset suhteet. Riippuvuus merkitsee, että jokin vaatimus on toteutettava toimiakseen ennen toista vaatimusta. Tuotteenomistaja määrittää sidosryhmien tuella asiakkaan ja liiketoiminnan saaman hyödyn vaatimuksille. Kehitystiimi määrittää työmäärän jokaiselle vaatimukselle. Scrum-tiimit käyttävät usein Fibonacci arvion luomiseen. Vaatimukset voidaan pisteyttää suhteellisesti, jotta ne ovat vertailukelpoisia toisiinsa nähden. Valitaan vaatimus, josta kehitystiimissä on sama näkemys ja käytetään tätä vaatimusten vertailukohtana. Määritetään muut vaatimukset ja päätetään, onko se muita vaatimuksia suurempi tai pienempi työmäärältään.

Kun vaatimukset on arvioitu, voidaan määrittää kunkin vaatimuksen suhteellinen prioriteetti. Suhteellinen prioriteetti auttaa ymmärtämään, miten tärkeä vaatimus on toisiin vaatimuksiin nähden. Kun tiedetään vaatimusten suhteellinen prioriteetti, niin voidaan muodostaa työlista seuraavaan iteraatioon ratkaisun tiekartasta. Suhteellisen prioriteetin voi laskea kaavalla *arvo / työmäärä*. Työlistan priorisoinnissa voi käyttää seuraavia kysymyksiä myöskin tukena.

- Mikä on vaatimuksen suhteellinen prioriteetti?
- Mitkä ovat vaatimuksen edellytykset?
- Mitkä vaatimukset kuuluvat yhteen ja muodostavat toimivan kokonaisuuden?

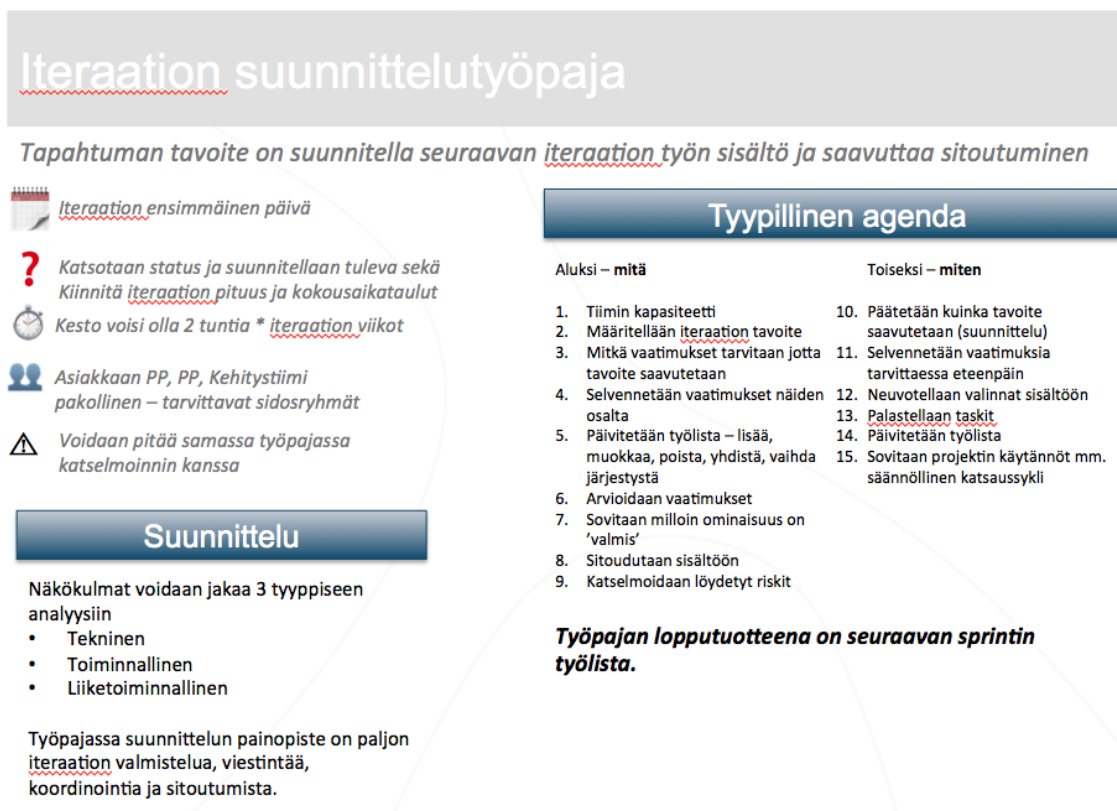
Priorisoidun käyttäjätarinoiden listaa kutsutaan ratkaisun työlistaksi. Ratkaisun työlista on tärkeä ketterän menetelmän asiakirja, artefakti. Kun ratkaisun työlista ja tiekartta ovat valmiita voidaan alkaa suunnitella iteraatioita osana ratkaisun kokonaissuunnitelmaa.

Vaikka ketterät projektit ovat joustavia monin tavoin sekä kehitystiimille, että loppukäyttäjille, suunnittelua tarvitaan vielä ennen työn aloittamista. Ketterän projektin ensimmäistä viikkoa kutsutaan usein nimellä Iteraatio 0 tai lämmittelyiteraatio. Tämän ajanjakson tavoitteena on aloittaa projekti. Asiakkaan projektipäällikkö toimii keskeisenä yhteyshenkilönä liiketoimintahenkilöiden ja kehitystiimin välillä alkaen ensimmäisestä iteraatiosta aina projektin loppuun saakka. Iteraatio 0 aikana loppuasiakkaat määrittelevät ensisijaisen työn tai tuotesuunnittelun ja projektipäälliköt varmistavat resurssit ja kehitystiimiä koskevat tilat, työvälineet ja menetelmät ja luo tavoitteet ja rutiinit tulevalle työlle ympäristö- ja arkkitehtuurisuunnittelun avulla. Vaiheen aikana hankitaan alustava tuki, sitoutuminen ja rahoitus projektille sekä toimitaan aktiivisesti sidosryhmien kanssa mallin alkuvaiheessa. Työpajassa tehdään alustava ratkaisun ideointi ja suunnitelma sidosryhmien kanssa tunnistamaan järjestelmävaatimuksia ja työkaluja. Aktiivisen sidosryhmien osallistumisen edistämiseksi on käytettävä visuaalisia työkaluja, jotta suunnitelmat ja prototyypit voidaan tehdä mahdollisimman tehokkaasti.

Kehityshankkeen alussa on aloitettava avaintiimien ja -roolien jäsenten tunnistaminen ja aloitettava niiden kiinnittäminen tiimiin. Tässä vaiheessa on hyvä olla vähintään yksi tai kaksi avainkehittäjää, projektipäällikkö ja yksi tai useampi sidosryhmien edustaja.

Projektin alussa mallinnetaan järjestelmän alustava arkkitehtuuri. Tässä vaiheessa on oltava ainakin yleinen käsitys siitä, miten järjestelmää aiotaan kehittää. Projektin kehittäjät tulevat mielellään samaan tilaan, usein taulun ympärille, keskustelemaan ja piirtämään järjestelmän mahdollista arkkitehtuuria. Tämä arkkitehtuuri todennäköisesti muuttuu ajan myötä, sen ei tarvitse olla kovin yksityiskohtainen ja dokumentaatiota tuotetaan yleisellä tasolla. Arkkitehtuurin osalta arvioidaan mahdolliset työkalut, kuten tietokannat, ETL-työkalut, visualisointi, palvelimet jne. Tavoitteena on tunnistaa arkkitehtuuristrategia. Kehitysvaiheiden aikana suunnitelmat ja tietomallit viedään tarkemmalle

tasolle. Ympäristön käyttöönottoon tarvitaan myöskin käyttäjäoikeudet, työasemat, kehitystyökalut, työskentelytilat tiimille.



Kuva 26. Iteraation suunnittelutyöpaja

Iteraation suunnittelutyöpajassa kootaan alustava arvio projektista, joka perustuu alku-peräisiin vaatimuksiin, arkkitehtuuriin ja tiimin taitoihin. Tämä arvio kehittyy koko hankkeen ajan. Kuvassa 26 on esitetty työpajan sisältö ja tavoitteet.

Iteraation 0 alkuvaiheessa kerätään loppukäyttäjiltä käyttäjätarinoita. Jokainen sisältää pienen toiminnallisen toiminnon, joka tuottaa käyttäjälleen sekä yritykselle arvoa. Jokaisesta tarinasta usein riittää, että kirjoitetaan vain muutama lause. Tarinan nimen tulee olla ominaisuutta kuvaava. Käyttäjätarinassa kerrotaan, kuka tarvitsee vaatimuksen, mitä se tekee ja miksi sitä tarvitaan. Vesiputousprojektissa määrittelijä usein dokumentoi vaatimuksen erittäin yksityiskohtaiselle tasolle. Analyysi tapahtuu usein kuukausia ennen kuin kehittäjä toteuttaa vaatimusta, joten sen on oltava kattava, jotta kaikki yksityiskohdat muistetaan toteutusta tehdessä. Käyttäjätarinassa on tärkeää, että asiakkaiden edustajat, kehittäjät ja testaajat toimivat yhdessä tarinan kanssa. Kun ongelmia ilmaantuu niin määrittelyt ovat tuoreina kaikkien mielissä ja yksityiskohdat vaatimuksista voidaan selkeyttää nopeasti lyhyellä keskustelulla asiakkaan kanssa. Tämä tarkoittaa myös sitä, että asiakkaiden on työskenneltävä tiiviisti kehitystiimin kanssa. Asiakkaan edustajat eivät pysty viettämään vain muutamaa viikkoa projektin alussa ja sitten unohtamaan pitkäksi aikaa projektia, kunnes ratkaisu testataan tai toimitetaan. Pidemmällä aikavälillä yksityiskohtaiset kuvaukset vaatimuksista on tosiasiallisesti toteutettu toimitetuissa

ratkaisuissa ja ratkaisun tekninen toimivuus testattu automatisoiduissa testeissä. Mikäli asiakas havaitsee puutteita ratkaisussa, niin toteutetaan puutteet ketterän prosessin iteraatiivisuutta hyödyntäen seuraavassa iteraatiossa ja ratkaisu vastaa liiketoiminnan nykyisiä vaatimuksia.

Toisin kuin perinteiset vaatimukset käyttäjätarinat ovat aikapohjaisia - kasvavat, kehittyvät ja erääntyvät ajan kuluessa tuotannossa, iteroinnissa ja päivittäisessä suunnittelussa. Käytettävissä on useita tekniikoita, joiden avulla voidaan määritellä käyttäjätarinoita. *INVEST*-mallin periaatetta voidaan käyttää kirjoitettaessa hyvää käyttäjätarinaa.

Independent/Itsenäinen - Yhden käyttäjätarinan tulee olla riippumaton toisesta tarinasta niin paljon kuin mahdollista. Suunnittelun, priorisoinnin ja arvioinnin väliset riippuvuudet tarinoista vaikeuttavat paljon.

Negotiable/Neuvoteltavissa - Tarinan tiedot voidaan laatia iteraation suunnittelukokouksen aikana. Tarina, jossa on liian paljon yksityiskohtia, voi rajoittaa keskustelua.

Valuable/Arvokas – Ratkaisu tuottaa arvoa asiakkaalle.

Estimable/Arvioitavissa - Kehittäjille on annettava riittävästi yksityiskohtaista tietoa, jotta voidaan arvioida käyttäjätarina ja näin ollen tarinan priorisointi ja suunnittelu olisivat mahdollisia.

Small/Pieni - Hyvän tarinan pitäisi olla pieni tyypillisesti enintään 2-3 henkilötyöviikkoa työtä.

Testable/Testattava - Käyttäjätarinoiden on oltava testattavissa tietyin hyväksymisperustein.

Vaatimusten kerääminen on kulmakivi minkä tahansa ketterän projektin onnistumiselle. Sidosryhmät usein kamppailevat toiminnoista ja ominaisuuksista, joita he odottavat järjestelmästä samanaikaisesti. Ketterissä projekteissa käytetään usein käyttäjätarinoita, jotka toimivat lähtökohtana vaatimusten kehittämisen, suunnittelun, riskien ja kustannusten edelleen tarkentumiselle. Organisaatioiden tulee investoida työkaluihin, joilla tuetaan ketterää työskentelytapaa. Työkaluja tulee käyttää eri tiimien liittämiseksi toisiinsa. Työkaluja voidaan käyttää tietojen jakamiseen, ajan tasalla pitämiseen, seurantaan ja ohjeistuksiin sekä pitää yllä hankkeen edistymistä. Yhteisten kommunikaatiovälineiden käyttäminen johtaa reaaliaikaiseen tietojen jakamiseen ja parempaan avoimuuteen ja näkyvyyteen työmääristä ja viime kädessä aikataulusta. Kuitenkin vaatimusten käsittely on helpommin sanottu kuin tehty. Asianomaisten sidosryhmien erilaisia työkaluja ja tekniikoita on ehkä sovellettava. Toisin kuin perinteisessä hankkeessa, jossa vaatimusten käsittelyvaihe koostuu yksityiskohtaisesta analyysistä ja täydellisestä määrittelydokumentista. Ketterä hanke pohjimmiltaan aluksi keskittyy korkean prioriteetin ominaisuuksiin ja integroituu vähitellen yksityiskohtiin. Tarinataulut ja prototyyp-

pit ovat tehokkaita tapoja kerätä palautetta ja sisällyttää ominaisuudet nopeasti ratkaisuun. Käyttäjien mahdollisuus nähdä ominaisuudet aikaisin toimivassa ratkaisussa vähentää jälkikäsitteilykustannuksia, koska esimerkiksi integraatiota muihin järjestelmiin ei ole vielä tehty.

Työpajat edistävät ryhmien välistä yhteistyötä. Yhteistyö ja avoimuus ovat tärkeitä näkökulmia ketterissä menetelmissä. Ajatellaan skenaariota, jossa tilaaja antaa odotuksia järjestelmän käyttäytymisestä, mutta jättää järjestelmän suunnittelun näkökohdat toimitajan teknisille asiantuntijoille. Työpaja yhdistää nämä ryhmät. Jokainen omassa siiossaan työskentelevä ryhmä on katastrofiriski. Avoimuuteen tarvitaan viestintäkanavia, jotka johtavat korkeaan luottamustasoon. Vaatimusten dokumentointi voi olla missä tahansa muodossa - laskentataulukot, asiakirjat, esitykset, kaaviot ja käyttäjätapausten käyttö. Eteenpäin ajamisen voima ei ole muoto, vaan sisältö (mitä), konteksti (missä) ja yhteistyö eri tiimien välillä. Avoimuus auttaa myös keräämään, kehittämään ja erottamaan "must have" vaatimukset "good to have" -vaatimuksista varhaisessa vaiheessa. Näitä tietoja voidaan käyttää priorisointiin iteraatioissa ottaen huomioon liiketoiminnan käyttäjätarpeet ja arvot sekä muutoksenhallinnan näkökulma.

Koska vaatimusprosessi on ketterä, hankkeet voivat saavuttaa arvontuottoa tehokkaammin, kun työvaiheet toimivat rinnakkain sen sijaan, että odottavat täysin määriteltyjä vaatimuksia, joilla voi olla vähän vuorovaikutusta sidosryhmien välillä. Hyötyjä on paljon verrattuna perinteiseen vaiheittaiseen malliin. Tämä varmasti merkitsee suurta muutosta siihen, miten eri organisaatioiden ratkaisuja suunnitellaan ja toimitetaan lyhyemmässä ajassa.

KETTERÄ TIIMI

Yksi keskeisistä näkökohdista on yritysten tietoisuus kehitystiimin hallinnasta. On havaittu, että kehitystiimit, sidosryhmät ja projektiorganisaation jäsenet työskentelevät organisaation ekosysteemeissä. Hankkeissa on tavallisesti myös muita osapuolia työskentelemässä kehitystiimin kanssa ja tiimien välinen yhteistyö hyödyttää molempia osapuolia avoimen toimintakulttuurin ja yhdessä tekemisen kautta. Organisaatiossa voidaan työskennellä liiketoiminnan tai teknisten näkemysten suhteen, joita kehitystiimin tulisi edistää. Yhteistyö voi vähentää kokonaistoimitusaikaa ja kustannuksia hyödyntämällä olemassa olevaa osaamista tai luomalla uutta osaamista vastaamaan yritystason strategiaa, jota tulevat hankkeet hyödyntävät uudestaan. Toisin sanoen tiimit voivat viettää vähemmän aikaa samojen asioiden uudelleen tekemiseen ja keskittyä tuottamaan todellista lisäarvoa sidosryhmilleen. Lisäksi, työskentelemällä läheisesti yritysten ammattilaisten kanssa, tiimit voivat helpommin mennä oikeaan suuntaan liiketoimintatarpeita huomioiden. Kolmanneksi se lisää mahdollisuutta, että toimittaja auttaa optimoimaan koko organisaation toimintaa, eikä pelkästään ratkaisua, jonka parissa työskennellään. Koska Leanilla kehitystoiminnalla on tarkoituksenmukainen näkemys, se lisää tiimin tehokkuutta ja näin ollen julkaisu saadaan markkinoille nopeammin.

Ketterän tiimin tunnistettuja ominaisuuksia ovat.

- Yksilöiden tietoisuus. Tästä näkökulmasta kyse on siitä, miten joku voi kehittää itseään hankkimalla uusia taitoja, näkemyksiä, kokemuksia ja niin edelleen. Kohdeyrityksen työntekijät ovat pääosin moniosaajia ja usean taidot ovat kehittyneet erilaisten toimeksiantojen kautta. Yleensä alkuvaiheessa kokeneempi työtoveri on päävastuussa ja jonkin osa-alueen alkuvaiheessa oleva osaaja kehittää osaamistaan työparina. Tämä on tarpeen, kun projektien laajuus vaihtelee yhden henkilön työpanoksesta monen osaajan kehitystiimiin.
- Tiimin tietoisuus. Tässä keskitytään siihen, miten tiimi voi oppia ja parantaa yhdessä. Tämä on ollut ketterän yhteisön ensisijainen tavoite. Ratkaisuja kehittävät tiimit, joten lisäämällä tiimiin vastuuta ja tietoisuutta kehittäjät pystyvät parantamaan kokonaistuottavuuttaan. Tiimit eivät tee työtä jota ei tarvita tai ei vie ratkaisun kehitystä eteenpäin.
- Yritysten tietoisuus. Ihmiset voivat ottaa huomioon organisaationsa yleiset tarpeet ja varmistaa, että heidän tekemällään toiminnalla edistetään organisaation tavoitteiden saavuttamista eikä pelkästään heidän ryhmänsä tavoitteiden saavuttamista.

Ketterä tiimi on itseohjautuva ja sillä on perinteisiä menetelmiä suuremmat mahdollisuudet tehdä luovia ratkaisuja päästääkseen haluttuun ratkaisuun eli välttämättä ei ole erikseen määrittelijää, konseptioijaa, kehittäjää vaan ne voivat olla sama henkilö. Kun vastuu projektin työvaiheiden lopputuloksesta on tiimillä, niin tiimillä on myös valta tehdä päätöksiä. Itseohjautuva kehitystiimi ei tarkoita, ettei tiimillä olisi projektinhallintaa vaan tiimiä, joka kykenee vastaamaan muutoksiin ja etsimään onnistuneen ratkaisun kannalta tarvittavat vaihtoehdot. Muutostarve syntyy eri sidosryhmien muuttuvista ja ajan kuluessa tarkentuvista tarpeista. Kehitystiimin mahdollisuuksiin optimaalisen ratkaisun tuottamiseen luotetaan organisaatiossa. Näin myöskin vahvistetaan kehitystiimin jäsenten yksilöllisten, ketteriin menetelmiin tarvittavien tietojen, taitojen ja luonteenpiirteiden merkitystä. Kohdeyritys onkin rekrytointiprosessissaan painottanut palkattavan henkilön luonnetta sekä karismaa. Ketteryys korostaa organisaation kyvykkyyttä luottaa kehitystiimin kompetenssiin sekä mahdollistaa päätöksenteon tiimille näkyväksi. Ketterät kehitysmenetelmät korostavat projektin aikaista, todelliseen palautteeseen pohjautuvaa muuttuvaakin määrittelyprosessia. Kehitystiimillä on mahdollisuus käyttää luovuuttaan sekä osaamistaan yhteisen tavoitteen saavuttamiseksi kattavan määrittelydokumentaation sijaan. Tiimin jäsenet ovat moniosaajia ja kantavat yhteisen vastuun tiimin menestyksen eteen. Erikseen ei nimetä arkkitehtiä, kantaosaajaa, raportointiosaajaa vaan vastuut jaetaan osaamisen tai osaamisen kehittämistavoitteiden mukaisesti. Myöskin asiakkaan tiiviissä – päivittäisessä – yhteistyössä oleva henkilö katsotaan tiimin jäseneksi.

Tiimin roolit Vastuut ja velvollisuudet			
	Määrittely	Iteraatio	Testi
Asiakkaan PP	Omistaa vision, proto, työlistä, budjetti, ryhmän asettaminen	Jatkuva työlistan hallinta, käyttäjätarinoiden hyväksyntä	Julkaisun hyväksyntä, osana <u>retroa</u>
Tiimi	Työmääräarvio, perustaa ympäristöt ja kehitysvälineet, tukee ominaisuuksien luontia, tukee valmiin määrittelmää	Arvioi käyttäjätarina iteratiion suunnittelu, kokoukset, luo toteutuksen	Suorittaa yksikkötestit, toimittaa julkaisumuistion, tukee hyväksymis <u>testausta</u>
Eximian PP	Työtilat ja hallintomallit, asiakasvastuu, määrittää prosessi detaljit, luo valmiin määrittelmä	Tukee tiimiä ja auttaa saavuttamaan tavoitteet, poistaa esteet	Raportoi edistymisen, vikalista

Kuva 27. Projektitiimin roolit, vastuut ja velvollisuudet

Kohdeyrityksen hankkeissa on tunnistettu kolme erilaista avainroolia, jotka tarvitaan sitoutumisen ja yhteistyön sujuvuuden kannalta riippumatta projektin koosta tai kestoista. Vastuut ja velvollisuudet on jaettu kolmeen vaiheeseen määrittely, iteraatio ja testaus. Tiimin roolit, vastuut ja velvollisuudet on esitetty kuvassa 27.

Asiakkaan projektipäällikkö on yhdyshenkilö toimittajan kehitystiimin ja tilaajaorganisaation välillä. Hänen tehtäviin kuuluvat määrittely vaiheessa vastuu ja omistajuus hankkeen visiosta, kehitystiimin alkuvaiheessa tuottamasta prototyypin sisällöstä, tulevan iteraation työlistan priorisoinnista. Sidosryhmien sitouttaminen ja sponsorin saaminen hankkeen tueksi on tärkeä, jotta hankkeen onnistunut läpivieminen taataan. Hän myöskin asettaa ryhmän, jolla varmistetaan, että oleellisten liiketoimintatietojen saaminen projektiin on varmistettu ja ryhmän henkilöillä on muiden tehtäviensä ohella riittävästi aikaa kehityshankkeelle. Iteraation aikana asiakkaan projektipäällikkö ylläpitää ja päivittää säännöllisesti työlistaa ja tuo uusia vaatimuksia käyttäjätarinoiden muodossa työlistalle. Hän hyväksyy asiakkaan puolelta mitkä ominaisuudet tulevat työlistan kautta priorisointiin ja harkintaan seuraaviin iteraatioihin. Testausvaiheessa hän hyväksyy julkaisun ja varmistaa, että toiminnallisuudet ovat hyväksymiskriteerien mukaiset ja voidaan todeta julkaisu valmiiksi tehdyksi.

Kehitystiimi määrittelyvaiheessa arvioi ja antaa vaatimuksille työmääräarviot. Kehitystiimin vastuulla on myös perustaa tai määritellä kehitysympäristöjen kapasiteetti, työvä-

lineet, ohjelmistot, ympäristöjen väliset yhteydet ja ympäristön käyttöönotto. Jäsenet ovat myös aktiivisesti mukana, kun uusia ominaisuuksia ja vaatimuksia pohditaan ja antavat tarvittaessa liiketoiminnasta tietämyksen ja teknisten vaatimusten tuen päätöksentekoon. Vuorovaikutus sidosryhmien kanssa on myös tärkeää, kun ominaisuuksille määritellään valmiin määritelmä sekä hyväksymiskriteerit. Iteraation aikana tiimin jäsenet tukevat uusien käyttäjätarinoiden arviointia, suunnittelevat ominaisuuksien toteuttamisen vaihtoehtoja sekä osallistuvat aktiivisesti määriteltyihin kokouksiin ja toteuttavat ratkaisun. Testivaiheessa kehitystiimi tekee yksikkö- ja integraatiotestauksen. Julkaisumuistio luodaan ennen käyttöönottoa, jotta kaikki ratkaisun uudet, jatkokehityt ja korjatut ominaisuudet tulevat loppukäyttäjien tietoisuuteen. Hyväksymistestauksessa kehitystiimi on mahdollistajana eli vie ratkaisun tuotantoon ja valmistele kaikki ratkaisun tarvitsemat yhteydet jne. Kehitystiimin tehtävänä on myös jatkuvasti kehittää omaa osaamistaan sekä kerätä parhaita käytäntöjä, jotta varmistetaan tiimin kilpailukyky. Uudet teknologiat ja työkalujulkaisut sekä versiopäivitykset huolehditaan kehitystiimin toimesta. Myöskin asiakkaan haluama taso dokumentoinnin ja työohjeiden sekä käyttöohjeiden osalta tuotetaan iteraatioiden aikana.

Toimittaja asettaa myös projektipäällikön, joka vastaa työtiloista, hallintomallista, projektissa käytettävistä menetelmistä sekä kommunikaatiokäytännöistä. Toimittajan projektipäällikkö on asiakaskommunikaation ja raportoinnin vastuuhenkilö sekä tukee asiakkaan projektipäällikköä vision, työlistan, hyväksymiskriteerien ja muiden tarvittavien tehtävien tukihenkilö. Projektipäällikön vastuulla on tukea kehitystiimiä kaikin tavoin ja poistaa esteet työn edistymisen osalta. Rooliin kuuluu myös edistymisenseuranta ja raportointi sidosryhmille sekä vikalistan ylläpito ja jatkotoimenpiteet.



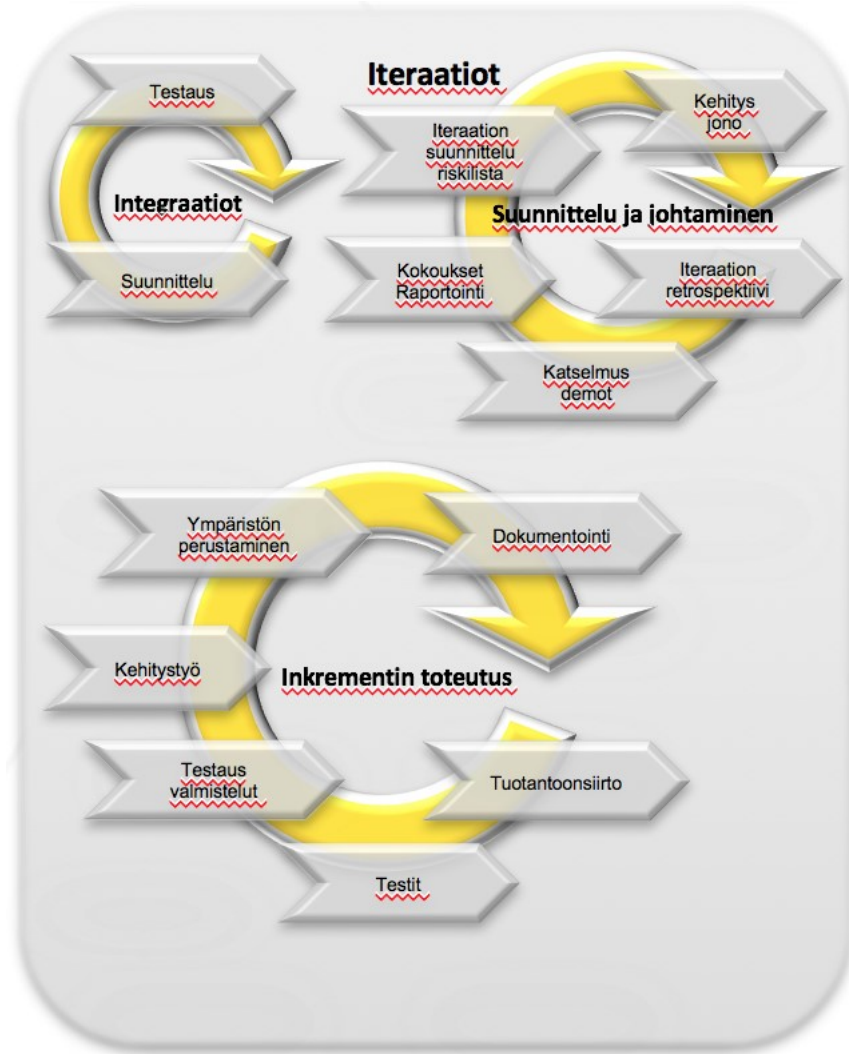
Kuva 28. Projektiorganisaatio

Projektiorganisaatio määräytyy hankkeen laajuuden, sisällön ja asiakkaan vaatimusten mukaan. Projektiorganisaatiossa olevia rooleja on sidosryhmät, sponsori, hankkeen ohjausryhmä, projektipäällikkö ja ketterä kehitystiimi. Projektiorganisaatio kuvattu kuvassa 28.

Projektiorganisaatiota skaalataan hankkeen tarpeen mukaan. Projektin tarvitsemiin osajiin vaikuttavat myös projektissa tarvittava osaaminen. Kehitystyö voi kohdistua esimerkiksi tietokantoihin, tiedonsiirtoihin, raportointeihin tai kaikkiin edellä mainittuihin tehtäviin. Rooleja ovat esimerkiksi tietokanta-asiantuntija, ETL-kehittäjä, raportointikehittäjä, konseptioija, business-analyytikko, web-kehittäjä jne.

5.3 Iteraatiot

Iteraatio on aikarajattu sykli, jonka tavoitteena on tuottaa toimiva ratkaisu syklin lopussa. BI-hankkeissa voi olla tehtäviä laidasta laitaan liittyen tietokantaympäristön perustamisesta, tietomallista raportoinnin kehittämiseen. Kehitystehtävien osa-alueet voidaan jakaa kuvan 29 esittämällä tavalla vaiheittain, mikäli projektiin kuuluu useita kehittämisen osa-alueita. Ympäristöön liittyviä ovat tekninen arkkitehtuurisuunnittelu, ympäristöjen ja työkalujen asennus sekä konfigurointi ja tarvittavien yhteyksien rakentaminen integroitavien järjestelmien välillä. Mallinnukseen vaikuttaa tietomallin valinta ja suunnittelu. Malleja voivat olla esimerkiksi Dimensionaalinen tai Data Vault. Iteraatioon sisältyy myös työtehtäviä, kuten ETL-toteutus ja fyysinen tietomalli, suunnittelu ja tietokannan perustaminen, visuaalisten ratkaisujen suunnittelu – raportit, portaalit ja työpöydät. Useissa asiakastoimeksiannoissa on myös tehtäviä liittyen kehitystä tukevien työkalujen asennus ja käyttöönotto. Iteraatiot voivat sisältää eri alueiden valmistelu-, määrittely-, integraatio-, johtamis-, suunnittelu- ja toteutusaiheita.



Kuva 29. Iteraatiot

Iteraation kohdeyritys jakaa kuuteen vaiheeseen, joita ovat prototypointi, määrittely ja suunnittelu, toteutus, testaus, hyväksymistestaus ja käyttöönotto.

Prototypointi

- Prototypointi on nopeasti ja kustannustehokkaasti tehtyjä karkean tason malleja suunnitellusta ratkaisusta. Niiden avulla testataan ja varmistetaan, että kehitetty konsepti vastaa asiakastarpeisiin, on käyttäjilleen ymmärrettävä ja arvoa tuottava.
- Saadun palautteen avulla ratkaisua voidaan kehittää ja parantaa ennen palvelun lopullista ratkaisua ja julkaisua.
- Prototyyppejä tehdään usein useampi iteraatio, joiden kautta suunnitelma tarkentuu ja saa halutun sisällön.
- Prototypointi on myös tapa minimoida kehityshankkeen epäonnistumisen riskiä.

Määrittely ja suunnittelu

- Arvioidaan toiminnalliset vaatimukset, miten ratkaisun tulisi toimia.
- Analysoidaan toimintaympäristö, toteutusympäristö, suorituskyky-, luotettavuusvaatimukset.
- Analysoidaan ja jäsennetään halutun toiminnallisuuden käsitteistö ja malli eli yksinkertaistettu kuvaus.
- Luodaan ja suunnitellaan priorisoitu työlista.
- Arvioidaan tekninen ympäristö, evaluoidaan ja valitaan uudet tuotteet.
- Muodostetaan ja estimoidaan priorisoitu työlista, joka iteraation aikana toteutetaan.
- Laaditaan ja hyväksytään kustannus- ja työmääräarvio sekä riskilista.

Toteutus

- Suunnitellaan ja toteutetaan tietomalli.
- Asennetaan ja käyttöön otetaan tarvittavat BI-työkalut.
- Toteutetaan tietomalli ja lataukset sekä ladataan tiedot tietomalliin.
- Luodaan tarvittavat raportit, dashboardit, näytöt valituilla työkaluilla ja vaatimusten mukaisilla sisällöillä.
- Testataan toteutus käyttäjien kanssa demokatselmuksessa.
- Identifioidaan ja kehitetään tukitoimet, joita saatetaan tarvita, kun ratkaisu on käyttöön otettu.

Testaus ja hyväksymistestaus käsitellään omassa kohdassa 5.4 testit ja hyväksyntä sekä käyttöönotto kohdassa 5.5.

Näkökulma	Johtaminen	Projekti	Kehitys
Keskity	Budjetti ja kontrolli	Henkilöstöjohtaminen ja edistymisen seuranta	Artifaktien luonti ja ratkaisun toteutus
Ketterä	Julkaisun työlista – teemat ja tavoitteet	Ratkaisun työlista - käyttäjätarinat	Iteraation työlista – tarinoiden ominaisuudet
Kesto	1-3 työpäivää	Iteraatio 0 1-2 viikkoa	Iteraatio 1-n 4-n viikkoa
Työkalut	Liiketoiminnan tavoitteet	Yhteisymmärrys BI ja raportointi Tietokanta Datan profilointi	Tietomalli BI ja raportointi ETL Datan laatu Master Datan hallinta Tietokanta Testaus
Tulokset	Korkean tason arviot Visio ja tiekartta	Liiketoimintamalli Liiketoiminnan termit Liiketoiminnan määrittäminen Prototyyppi Datan profiilit Tarinakortit Iteraation tiekartta	Tietokannat ja kuutiot ETL Työpöydät ja raportit

Kuva 30. Iteraation näkökulmia BI-kehityksessä

Näkökulmia iteraatioon on esitelty kuvassa 30. Iteraatiota voidaan tarkastella johtamisen, projektin ja kehityksen näkökulmista. Tekijöiksi olen valinnut fokuksen, ketteryyden, toimintaa ohjaavat työkalut sekä tulokset.

5.4 Testit ja hyväksyntä

Testaus on ratkaisun suorittamista tavoitteena löytää virheitä. Testauksen pohjimmaiset päämäärät voidaan jakaa kahteen osaan, verifiointiin ja validointiin sekä laadun varmistukseen. BI-kehityksessä testauksen tarkistuspisteet voidaan jakaa datan keräämiseen, datan integrointiin, datan varastointiin ja datan esittämiseen. Kohdeyrityksen testausprosessi kehittyi osana tätä tutkimusta. Edellä mainituissa tarkistuspisteistä on esitelty hyvät käytännöt, jotka yleisesti tulevat vastaan hankkeissa. Testausvaiheet on jaettu yksikkö-, integraatio-, suorituskyky- ja hyväksymistestaukseen. Kohdeyrityksen testausprosessi on esitelty alla olevassa kuvassa 31.



Kuva 31. Testausprosessi

Testaussykli on jaettu seuraaviin osatekijöihin liiketoiminnan ymmärtäminen, testaus suunnitelman laatiminen, testitapausten luonti, testiympäristön luonti ja konfigurointi, testidatan luonti, testaus, testitulokset ja analyysi. Näiden vaiheiden suunnittelun ja toteutuksen myötä luodaan toimiva ratkaisu.

Hyväksymistestaukseen osallistuvat kehitystiimi ja asiakkaan liiketoiminnan edustajat. Projektin työlistan aihioille on määritelty hyväksymiskriteerit, joiden tulee toteutua, jotta ratkaisun katsotaan olevan hyväksytty julkaistavaksi ja käyttöönottoon. Hyväksymiskriteerit on tarkistuslista, joka määrittää vaatimusten hyväksymisvaatimukset. Hyväksymiskriteerit ovat yksiselitteiset, mielellään mitattavat kriteerit, joiden täytyttyä kyseinen työlistan kohta voidaan todeta valmiiksi. Hyväksymiskriteerit eivät välttämättä

kuvaa kaikkia mitattavia ominaisuuksia, vaan nostavat esille hyväksynnän kannalta keskeiset seikat, eräänlaisen minimin riman ylitykselle.

5.5 Julkaisu

Kun hyväksymiskriteerit on kaikilta osin hyväksytty, suoritetaan ratkaisun julkaisu ja käyttöönotto. Käyttöönoton aluksi valmistellaan käyttöympäristö kaikilta osin - palvelimet, työkalut ja sisältö sekä mahdolliset yhteydet tietojärjestelmiin. Loppukäyttäjät aloittavat sovitun pituisen pilotointijakson ja järjestelmätestausta sekä hienosäätöä suoritetaan todellisessa tuotanto- ja käyttöympäristössä. Projektin toimituksen osana voi olla ratkaisun käyttöohjeet ja alussa aloitetaan tarvittaessa käyttäjien koulutus ja tiedottaminen ratkaisun ominaisuuksista. Ratkaisusta riippuen suoritetaan myös lähtevän tiedon testaaminen, siirretyn tiedon jatkokäytön testaaminen sekä saapuvan tiedon testaaminen. Tiedonsiirtojen osalta tehdään mahdollisesti historiadatan osalta sovitun aikavälin datan tiedonsiirto, jonka jälkeen aloitetaan reaaliaikaiset tiedonsiirrot tai sovitun aikavälin jaksoissa uudet ja muuttuneet sekä poistettujen tietojen päivitykset. Tämän jälkeen voidaan koko ratkaisulle suorittaa suorituskäytöstä tuotantoympäristössä. Käyttöönoton jälkeen jatketaan käyttäjien tukitoimia sovitun sopimuksen mukaisella mallilla ja päivitetään jatkokehitystoiveet työlistaan.

5.6 Projektin päätös

Projektin päätöksessä pidetään jatkuvan parantamisen mallin mukaisesti retrospektiivi, jossa käydään läpi mikä meni hyvin, asiat jotka olisivat voineet mennä paremmin, asiat jotka yllättivät, opitut asiat ja ajatukset, joita projektin aikana syntyi. Nämä asiat kirjataan projektin jälkeen lomakkeelle, jossa on jokaisen edellä mainitun asian osalta otettu huomioon liiketoiminta, vaatimukset, prosessi, johtaminen ja teknologia. Ajatukset osiossa kirjataan mielipide mitä asioita tiimi haluaa jatkossa pitää ja mitä muuttaa. Kuvassa 32 on esitetty iteraation katselmointityöpajan tavoitteet ja agenda. Mallissa on mahdollista myös pitää samassa kokouksessa iteraation suunnittelu- ja katselmointityöpajat. Yleensä näin toimittaessa toimittajan ja asiakkaan yhteistyö on jatkunut useita projekteja ja näin ollen organisaation henkilöt ja ympäristöt sekä toimintatavat tunnetaan riittävästi hyvin.

Iteraation katselmointityöpaja

Tapahtuman tavoite on esitellä mitä iteraation aikana saavutettiin ja ratkaisun nykytila



Iteraation viimeisenä päivänä



Esitellään saavutukset, keskustellaan ratkaisusta, työlistasta sekä tulevat tavoitteet



*Kesto voisi olla 1 tuntia * iteraation viikot*



*Asiakkaan PP, PP, Kehitystiimi
pakollinen – tarvittavat sidosryhmät*



*Voidaan pitää samassa työpajassa
suunnittelun kanssa*

Tyypillinen agenda

1. Esitellään toimiva ratkaisu projektiryhmälle ja sidosryhmille
2. Esitellään kun on tehty ylimääräisiä tarinoita
3. Keskustellaan jos iteraatiossa keskeneräisiä tarinoita
4. Päätetään iteraatio ja hyväksytään iteraation tulokset hyväksymiskriteerien pohjalta jos ei ole jo hyväksytty
5. Kerätään lista avoimista asioista tai ongelmista
6. Hoidetaan työlista – lisäykset, jakaminen, poistot
7. Priorisoidaan työlista seuraavaa iteraatiota varten

Kuva 32. Iteraation katselmointityöpaja

Projektin päätöksen jälkeen ratkaisun käyttäjä- sekä kehitystukea jatketaan loppukäyttäjille. Kohdeyrittäjä on käytössä myös takuu, joka sopimuksesta riippuen vaihtelee kestoltaan mutta pitää sisällään takuuaajan, jonka puitteissa havaitut puutteet tai virheet korjataan ilman lisäveloitusta. Projektin päätöksen yhteydessä pyritään myös sopimaan jatkokehityshankkeista sekä sovitaan asiakkaan kanssa projekti päättyneeksi.

5.7 Artefaktit

Kohdeyrittäjän artefaktit projekteissa hiukan vaihtelevat mutta pysyviä ketterissä hankkeissa ovat työlista, riskilista ja toimiva ratkaisu sekä sovituilta osin tuotettava dokumentaatio. Työlista sisältää priorisoidun listan kaikista vaatimuksista. Listaa päivitetään ja tarkennetaan jatkuvasti projektin edetessä. Ratkaisun työlistan avulla voidaan seurata projektin etenemistä ja hallita sen laajuutta. Työlista sisältää ratkaisun vision mukaisesti iteraatioon sisällytettävät asiakasvaatimukset, tarpeista syntyvät toiveet toiminnallisista piirteistä ja teknologiset vaatimukset ratkaisulle haluttujen piirteiden toteuttamiseksi sekä mahdollisesti julkaisuissa esiintyneiden virheiden korjaustehtäviä sekä tiedonsiirrot.

Riskilistan ja riskienhallinnan tavoitteena on tunnistaa ja hallita projektiin vaikuttavia potentiaalisia tekijöitä ja pitää ne sellaisena, ettei projektin toiminta, tavoite ja lopputulos ole uhattuna. Riskitekijöitä tunnistetaan kategorioissa ihmiset (saatavuus, moniosaaminen, allokointi), koko (suurta on vaikeampi hallita), prosessi (kuinka seurataan), teknologia/työkalut (uusi, kompleksinen), organisaatio sekä johtaminen (talous, tuki), asiakas (vaatimukset, muutokset, sitoutuminen), arviointi (resurssi, aika), myynti sekä tukitoimet (ymmärretään asiakas tarve). Riskienhallintaan käytetään ajatusta tunnistaa, analysoi, suunnittele ja seuraa. Riskilistaa ylläpidetään jatkuvasti ja riskejä arvioidaan aina henkilöiden välisissä tapaamisissa.

Viimeisenä ja tärkeimpänä toimituksena on toimiva ratkaisu, joka täyttää ja ylittää asiakkaalle luvatus ja asiakkaan odottaman arvon. Toimiva ratkaisu sisältää määritellyt vaatimukset ja ominaisuudet. Toimittajan ja asiakkaan välinen yhteistyö saattaa kestää useita projekteja ja vuosia, kun edellä mainitut tavoitteet saavutetaan. Tarkemmalla tasolla julkaisun laadullisia tekijöitä voivat olla ratkaisun toiminnallisuus, luotettavuus, käytettävyys, tehokkuus, ylläpidettävyys ja siirrettävyys.

6. POHDINTA JA YHTENNVETO

Tutkimuksen tavoitteena oli kehittää ketterä projektinhallintamenetelmä, jonka avulla organisaation, ja edelleen menetelmän käyttäjän on helpompaa koordinoita asiakasorganisaation BI-projekteja. Asiakasorganisaatiolla tarkoitetaan kaikkia kohderyhmän asiakkaita, joilla ei ole käytössään vakiintunutta projektinhallintamenetelmää tai haluavat toimittajan käyttävän omaansa. Tutkimusentekijä on työskennellyt kohdeyrityksen palveluksessa vuodesta 2004 lähtien ja kehitti mallin työnantajalleen. Projektinhallintamenetelmän tarve oli havaittu kohdeorganisaation projekteissa asiakkailleen. Kohdeorganisaatio, kollegat ja tutkimuksen tekijän kokemukset työssään määrittelivät pitkälle asetetut vaatimukset projektinhallintamenetelmälle. Kohdeorganisaation toimintaympäristö oli tutkimuksen tekemisen aikaan liiketoimintahallinnan järjestelmien konsultointi- ja kehityspalveluissa. Organisaation asiakkaat olivat suuria ja keskikokoisia kotimaisia sekä kansainvälisesti toimivia yksityisiä, että julkisia yrityksiä eri toimialoilta. Kohdeorganisaation toiminta perustui teknologioihin, monien eri toimittajien työkaluihin ja käytäntöihin suurten tietomassojen keräämiseen, integrointiin, analysointiin sekä visualisointiin asiakasorganisaatioille paremman päätöksenteon tueksi. Liiketoimintaa tukevien sekä lisäarvoa tuottavien ratkaisujen toimittaminen oli myös kohdeorganisaation tärkein asiakkailleen tuottama lisäarvo BI-projektien sekä BI-ratkaisujen avulla.

Ketterien projektinhallintamenetelmien hyödyntämisestä BI-projekteissa on saatavilla harmittavan vähän tutkittua tietoa. Tutkimustietoa ketteristä menetelmistä löytyy kuitenkin ohjelmistoprojekteista. Diplomityössä suoritettiin kattava ketterän menetelmän mallinnus BI-projektille, joka huomioi liiketoimintatiedonhallintaprojektien luonteenpiirteet. Saavutettujen tutkimustulosten pohjalta voidaan sanoa ketterien menetelmien soveltuvan hyvin myös BI-projekteihin, kun otetaan huomioon rinnakkaiset työvaiheet sekä mallilta edellytetty skaalautuvuus hankkeen koko luokasta riippuen. Ketterien menetelmien periaatteet ja käytännöt soveltuvat sellaisenaan BI-hankkeisiin.

Tuloksista voidaan poimia useita käytäntöä helpottavia suosituksia BI-projekteihin. Tutkimuksessa tarkasteltiin mallia johtamisen, projektinhallinnan sekä kehitystiimin näkökulmista. Projektin aloitusvaiheessa on hyvä käydä läpi asiakkaan toimintatapoja ja menetelmiä. Asiakkaan kanssa vuorovaikutuksessa löydetään heidän organisaationsa toimintatapoihin ja menetelmiin soveltuva menetelmä, joka yleensä sallii ketterän menetelmän piirteiden hyödyntämisen. Ketterien menetelmien etuina nousivat esiin seuraavat asiat. Empiirisuus eli kokeilujen ja havaintojen kautta etsitään jokaiseen hankkeeseen parhaiten sopivat käytännöt. Itseohjautuville tiimeille annetaan työrauha, josta seuraavat vastuut ja velvollisuudet hoidetaan sitoutuneesti sekä toimintaa jatkuvasti analysoiden ja kehittäen. Integraatioiden ja tiedonsiirtojen vaiheistaminen - riippuvuussuhteet huomi-

oiden – voidaan jakaa eri kehityspolkuihin. Työpaja käytäntö suunnittelu- ja hyväksyntävaiheissa auttaa yhteistyössä ja pitää toiminnan läpinäkyvänä. Työpajoissa myöskin asiat saadaan sovittua tehokkaasti yhteisymmärryksessä ja kaikkia osapuolia kuunnellen. Parhaassa tapauksessa suunnittelu- ja katselmointityöpajat järjestetään yhden päivän sisällä.

Ratkaisun testaamisen suhteen luotiin testausprosessi, joka yleensä ketterissä menetelmissä on kuvattu hyvin kevyesti, jos laisinkaan. Ratkaisun testauskäytännöt varmistavat onnistuneen ratkaisun kehittämisen ja antavat pohjaa laadullisille tavoitteille sekä vähentävät muutostöistä aiheutuvia kustannuksia ja lisätöitä. Malli velvoittaa myös riskilistan muodostamisen sekä löytyneiden tekijöiden kuvaamisen riskienhallinnan seurantaan luotuun riskilistaan. Artefaktit toimivat mallin dokumentaationa ja lisädokumentointia tehdään asiakasorganisaatiossa syntyvistä tarpeista, ei oletuksena.

Tutkimus onnistui vaiheiltaan erittäin hyvin ja kaikki osapuolet olivat sitoutuneita sekä kiinnostuneita aiheesta. Asiantuntijoiden apu mahdollisti eri näkökulmien huomioon ottamisen sekä kokemuspohjan käytännön sudenkuopista. Ketterälle mallille asetettuja tavoitteita vasten tarkasteltaessa tässä tutkimuksessa toteutettu ketterä malli onnistui siinä tarpeessa, johon se oli suunniteltu eli BI-projektin onnistuneeseen ketterään hallintaan kohdeyrityksen näkökulmasta. Hevner et al. (2004) esittelemään viitekehykseen pohjautuva projektinhallintamallin kehitys sekä analyyttinen arviointi saatiin toteutettua tutkimuksessa. Analyyttinen arviointi onnistui mittaamaan, täytyivätkö kohdeorganisaation asettamat vaatimukset mallissa.

Tutkimustulokset antavat myös mahdollisuuden tutkimuksen laajentamiselle. Kuinka kehitetty malli toimii, kun se käyttöön otetaan todellisiin projekteihin. Ketteriä menetelmiä, joita ei tässä tutkimuksessa käsitelty, mutta voisi jatkotutkia ovat esimerkiksi ”Large-Scale Scrum” skaalautuvuuden näkökulmasta. Kevyemmät kehityslähtöiset LeSS sekä Nexus olisivat uudempina malleina olleet hyviä tarkastelun kohteita.

LÄHTEET

- [1] Agile Alliance, List of Prerequisites for Starting Scrum in DW/BI Projects, 2015. Saatavissa: <https://www.scrumalliance.org/community/articles/2015/january/list-of-pre-requisites-for-starting-scrum-in-dwbi#sthash.E3Ldf7Tw.dpuf>
- [2] Agile Alliance, Manifesto for Agile Software Development, 2001. Saatavissa: <http://agilemanifesto.org/>
- [3] Agile Project Initiation Survey, 2013. Saatavissa: <http://www.ambysoft.com/surveys/projectInitiation2013.html#Downloads>
- [4] Ambler, W.S. & Lines, M.: Disciplined Agile Delivery: A Practitioner's Guide to Agile Software Delivery in the Enterprise. International Business Machines Press. Crawfordsville. 2012. Saatavissa: <http://library.books24x7.com.libproxy.tut.fi/toc.aspx?bookid=50660>
- [5] Ambler, W.S.: Construction goals, 2014. Saatavissa: <http://www.disciplinedagiledelivery.com/construction-goals-2/>
- [6] Ambler, W.S.: Disciplined Agile Delivery, 2014. Saatavissa: <http://www.disciplinedagiledelivery.com/introduction-to-dad/>
- [7] Ambler, W.S.: Full Agile Delivery Lifecycles, 2014. Saatavissa: <http://www.disciplinedagiledelivery.com/lifecycle/>
- [8] Ambler, W.S.: Inception goals, 2014. Saatavissa: <http://www.disciplinedagiledelivery.com/inception-goals-3/>
- [9] Ambler, W.S.: Ongoing goals, 2014. Saatavissa: <http://www.disciplinedagiledelivery.com/ongoing-goals-2/>
- [10] Ambler, W.S. 2014. Roles on DAD Teams. Saatavissa: <http://www.disciplinedagiledelivery.com/roles-on-dad-teams/>
- [11] Ambler, W.S. 2014. Transition goals. Saatavissa: <http://www.disciplinedagiledelivery.com/transition-goals-2/>
- [12] Cobb, C. G.: Making Sense Of Agile Project Management, Balancing Control & Agility. Wiley Publishing, 2011. Saatavissa: <http://ebookcentral.proquest.com/lib/tut/detail.action?docID=661483>
- [13] Davis, B.: Agile Practices for Waterfall Projects: Shifting Processes for Competitive Advantage, J. Ross Publishing, 2013. Saatavissa:

<http://library.books24x7.com.libproxy.tut.fi/assetviewer.aspx?bookid=58340&chunkid=479114061>

- [14] Davis, B., Radford, D.: *Going Beyond the Waterfall: Managing Scope Effectively Across the Project Life Cycle*, J. Ross Publishing, 2014. Saatavissa: <http://library.books24x7.com.libproxy.tut.fi/assetviewer.aspx?bookid=73143&chunkid=793824013>
- [15] Fitsilis P.: Comparing PMBOK and Agile Project Management software development processes. In: Sobh T. (eds) *Advances in Computer and Information Sciences and Engineering*. Springer, Dordrecht, 2008.
- [16] Girvan, L., Paul, D.: *Agile and Business Analysis*, BCS Learning & Development Limited, 2017. Saatavissa: <http://ebookcentral.proquest.com/lib/tut/detail.action?docID=4746867>.
- [17] Hevner, A.R., March, S.T., Park, J. & Ram, S. (2004). Design science in information system research. *MIS Quarterly*, 28(1), 75-105.
- [18] Hoogendoorn, S.: *This is Agile: Beyond the Basics. Beyond the Hype. Beyond Scrum.*, Dymaxicon, 2014. Saatavissa: <http://common.books24x7.com.libproxy.tut.fi/toc.aspx?bookid=71497>
- [19] Hovi, A., Hervonen, H., Koistinen, H.: *Tietovarastot ja Business Intelligence*, Jyväskylä, WSOYpro, Docendo, 2009. Saatavissa: <https://oukasrv6.ouka.fi/Intro?sesid=1450437552&formid=fullt&dat0=Tietovarastot+ja+Business+Intelligence&dat3=&dat4=&typ0=86&typ3=11&typ4=7&typ7=42>
- [20] Hughes, R.: *Agile Data Warehousing for the Enterprise: A Guide for Solution Architects and Project Leaders*. Morgan Kaufmann Publishers, 2016. Saatavissa: <http://common.books24x7.com.libproxy.tut.fi/toc.aspx?bookid=91943>
- [21] Hughes, R.: *Agile Data Warehousing Project Management: Business Intelligence Systems Using Scrum*, Morgan Kaufmann Publishers, 2013. Saatavissa: <http://library.books24x7.com.libproxy.tut.fi/assetviewer.aspx?bookid=51063&chunkid=937251817&rowid=13>
- [22] Juricek, J.: *Agile project management principles*, 2014. Paper presented at the, 2(2) 172. Saatavissa: doi:<http://dx.doi.org/10.7763/LNSE.2014.V2.117>
- [23] Järvinen, P. 2008. Mapping Research Questions to Research Methods. In D. Avidson et al., eds. *IFIP International Federation for Information Processing, Volume 274; Advances in Information Systems Research, Education and Practice*. Boston: Springer, ss. 29–41. Saatavissa:

http://link.springer.com.libproxy.tut.fi/chapter/10.1007%2F978-0-387-09682-7-9_3

- [24] Kimball, R., Ross, M.: The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling, Third Edition. John Wiley & Sons, 2013. Saatavissa: <http://common.books24x7.com.libproxy.tut.fi/toc.aspx?bookid=56391>
- [25] Kimball, Ralph. The Data Warehouse Lifecycle Toolkit, Second Edition. John Wiley & Sons, 2008. Books24x7. Saatavissa: <http://common.books24x7.com.libproxy.tut.fi/toc.aspx?bookid=24441>
- [26] Kimball, R., Ross, M.: The Kimball Group Reader: Relentlessly Practical Tools for Data Warehousing and Business Intelligence, John Wiley & Sons, 2010. Saatavissa: <http://library.books24x7.com.libproxy.tut.fi/assetviewer.aspx?bookid=34943&chunkid=438332002¬eMenuToggle=0&hitSectionMenuToggle=0&leftMenuState=1>
- [27] Knaster, R., Leffingwell, D.: SAFe 4.0 Distilled, Applying the Scaled Agile Framework for Lean Software and Systems Engineering, Addison-Wesley Educational Publisher Inc., 2017.
- [28] Layton, M., C.: Agile Project Management for Dummies, John Wiley & Sons, 2012. Saatavissa: <http://library.books24x7.com.libproxy.tut.fi/assetviewer.aspx?bookid=46057&chunkid=597176198&rowid=426¬eMenuToggle=0&leftMenuState=1>
- [29] Leffingwell, D.: Agile Software Requirements. Lean Requirements Practices for Teams, Programs, and the Enterprise. Addison-Wesley, 2011.
- [30] Leffingwell, D.: Responsibilities of Program Portfolio Management. Saatavissa: <https://www.scaledagileframework.com/program-portfolio-management/>
- [31] Leffingwell, D.: SAFe 4.0 Reference Guide: Scaled Agile Framework for Lean Software and Systems Engineering, Addison-Wesley Publisher Inc., 2016.
- [32] Leffingwell, D.: Scaling Software Agility. Best Practices for Large Enterprises. The Agile Software Development Series, 2007.
- [33] Leffingwell, D.: Welcome to Scaled Agile Framework V4.0!, 2016. Saatavissa: <https://www.scaledagileframework.com/about/>
- [34] Leffingwell, D., Widrig, D.: Managing Software Requirements: A Use Case Approach, Second Edition. Addison-Wesley Professional, 2003.

- [35] Measey, P.: Agile Foundation: Principles, Practices and Frameworks. BCS., 2015. Saatavissa:
<http://common.books24x7.com.libproxy.tut.fi/toc.aspx?bookid=64598>
- [36] Moran, A.: Managing Agile, Springer International Publishing, 2015. ProQuest Ebook Central. Saatavissa:
<http://ebookcentral.proquest.com/lib/tut/detail.action?docID=2094506>.
- [37] Moss, L., Atre S.: Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications, Addison-Wesley Information Technology Series, 2003.
- [38] Nicholas, J., M., Steyn, H., Project Management for Engineering, Business and Technology. Taylor & Francis, 2016. Saatavissa:
<http://ebookcentral.proquest.com/lib/tut/reader.action?docID=4790130&ppg=44>
- [39] Pham, A., T., Pham, P.V., Scrum in Action. Course Technology / Cengage Learning, 2011. Saatavissa:
<http://ebookcentral.proquest.com/lib/tut/reader.action?docID=3136417&ppg=43>
- [40] Pries, K., Quigley, J.: Scrum Project Management, 2011. Saatavissa:
<http://library.books24x7.com.libproxy.tut.fi/assetviewer.aspx?bookid=36985&chunkid=498903721&rowid=53>
- [41] Royce, W.: Managing the Development of Large Software Development and Enhancement, 1970. Proceedings, IEEE WESCON. Pages 1-9. Saatavissa:
<http://agileconsortium.pbworks.com/w/file/attach/52184636/waterfall.pdf>
- [42] Schiel, J., The ScrumMaster Study Guide, Auerbach Publications, 2012. Saatavissa:
<http://library.books24x7.com.libproxy.tut.fi/assetviewer.aspx?bookid=47196&chunkid=430096551&rowid=37¬eMenuToggle=0&leftMenuState=1>
- [43] Sherman, R.: Business Intelligence Guidebook: From Data Integration to Analytics, Morgan Kauffmann Publishers, 2015. Saatavissa:
<http://library.books24x7.com.libproxy.tut.fi/assetviewer.aspx?bookid=77362&chunkid=879646625¬eMenuToggle=0&hitSectionMenuToggle=0&leftMenuState=1>
- [44] Simon, A.: Modern Enterprise Business Intelligence and Data Management. Morgan Kaufmann Publishers, 2014. Saatavissa:
<http://common.books24x7.com.libproxy.tut.fi/toc.aspx?bookid=73422>
- [45] Simon H., A.: The sciences of the artificial, MIT Press, Cambridge, 1969.

- [46] Sommerville, I.: Software Engineering, 10. painos, Pearson Education, 2016.
- [47] West, D.: Water-Scrum-Fall Is The Reality Of Agile For Most Organizations Today. Forrester Research Inc., 2011.
- [48] Wysocki, R.: Effective Project Management: Traditional, Agile, Extreme, John Wiley & Sons, 2014. Saatavissa:
<http://library.books24x7.com.libproxy.tut.fi/assetviewer.aspx?bookid=62624&chunkid=649992775>